AÑO III - NÚM. 78

MICROHOBBY

SEMANAL

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

AÑO III - N.º 78

135 PTS.
Canarias 140 ptas.

MICROFILE

BASE DE DATOS PARA EL SISTEMA MDS

UTILIDADES

EL ESPÍA:

DESCUBRE Y MODIFICA LOS GRÁFICOS DE UN PROGRAMA

TOP SECRET

RUTINAS DE CARGA EN CÓDIGO MÁQUINA

NUEVO

CYBERUN

¡Acción y deporte al más alto nivel!

WINTER GAMES

JUSTICIEROS DEL SOFTWARE

EN BUSCA DEL PROGRAMA ESTRELLA

MOVIE, ACTUAL Nº 1



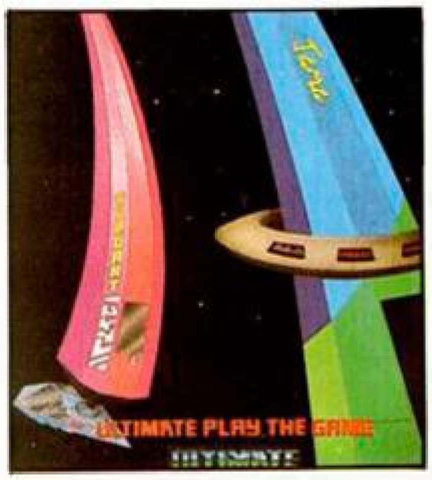
HOBBY PRESS


TU PROGRAMA DE RADIO claro!
88 96 100 104 108
700 800 1000 1200 1400 1600
SABADO CHIP
AUDISON2
• Entrevistas a fondo
• Exitos en Soft
• Noticias en Hard
• Concursos
Prográmatelo: Sábados tarde de 5 a 7 horas.
En directo y con tu participación.
LA COPE A TOPE.
—RADIO POPULAR 54 EMISORAS O.M.—
En Barcelona Radio Miramar
COPE rp

MICROHOBBY ESTA SEMANA

AÑO III. N.º 78. 13 al 19 de mayo de 1986.
135 ptas. Canarias, Ceuta y Melilla: 130 ptas.
Sobretasa aérea para Canarias: 10 ptas.





Consigue la fuente del Universo con el trepidante Cyberun. Pág. 12.

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo
Domingo Gómez

Asesor Editorial
Gabriel Nieto

Redactora Jefe
Africa Pérez Tolosa

Diseño
Rosa María Capitel

Redacción
Amalio Gómez, Pedro Pérez, Jesús Alonso

Secretaría Redacción
Carmen Santamaría

Colaboradores
Primitivo de Francisco, Rafael Prades, Miguel Sepúlveda, Sergio Martínez y J. M. Lazo

Corresponsal en Londres
Alan Heap

Fotografía
Carlos Candel

Portada
José María Ponce

Dibujos
Teo Mójica, F. L. Frontán, J. M. López Moreno, J. Igual, J. A. Calvo, Lóriga, J. Olivares

Edita
HOBBY PRESS, S. A.

Presidente
María Andrino

Consejero Delegado
José I. Gómez-Centurión

Jefe de Producción
Carlos Peropadre

Jefe de Publicidad
Concha Gutiérrez

Publicidad Barcelona
José Galán Cortés
Tels.: 303 10 22 - 313 71 76

Secretaría de Dirección
Pilar Aristizábal

Suscripciones
M.ª Rosa González
M.ª del Mar Calzada

Redacción, Administración y Publicidad
La Granja, 39
Polígono Industrial de Alcobendas
Tel.: 654 32 11
Telex: 49480 HOPR

Dto. Circulación
Paulino Blanco

Distribución
Coedis, S. A. Valencia, 245
Barcelona

Imprime
Rotedic, S. A. Ctra. de Irún, km 12,450 (MADRID)

Fotocomposición
Novocomp, S.A.
Nicolás Morales, 38-40

Fotomecánica
Grof
Ezequiel Solana, 16

Depósito Legal
M-36.598-1984

Representante para Argentina, Chile, Uruguay y Paraguay, Cia. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina)

Solicitado control OJD

MICROHOBBY NUMEROS ATRASADOS

Queremos poner en conocimiento de nuestros lectores que para conseguir números atrasados de MICROHOBBY SEMANAL, no tienen más que escribirnos indicándonos en sus cartas el número deseado y la forma de pago elegida de entre las tres modalidades que explicamos a continuación.
Una vez tramitado esto, recibirá en su casa el número solicitado al precio de 95 ptas. + 6 de IVA hasta el n.º 36, a 125 ptas. + 8 de IVA hasta el n.º 60 y a 135 ptas. desde el n.º 60 en adelante.



FORMAS DE PAGO

- Enviando talón bancario nominativo a Hobby Press, S. A. al apartado de Correos 54062 de Madrid.
- Mediante Giro Postal, indicando número y fecha del mismo.
- Con Tarjeta de Crédito (VISA o MASTER CHARGE), haciendo constar su número y fecha de caducidad.

RELACION AMSTRAD-SINCLAIR

UNA MEDIDA POLEMICA

La noticia que hace algunas semanas saltó a la opinión pública con respecto a la compra de Sinclair por Amstrad, aún no ha dejado de resonar en los medios especializados, y es por esta razón por lo que MICROHOBBY ha querido recopilar las opiniones de las compañías españolas que de alguna forma están relacionadas con esta nueva e importante circunstancia.

Hemos conversado con Investrónica, la distribuidora oficial en España de los productos Sinclair, con algunas de las más prestigiosas casas de software y con los comercios más importantes dedicados a la venta de ordenadores en general.

Las opiniones recogidas son de la más variada naturaleza, dependiendo cada una de ellas del enfoque que cada compañía puede dar a la misma circunstancia, pero prácticamente todas vienen a coincidir en que esta compra por parte de Amstrad va a influir favorablemente en muchos sentidos.

Las compañías relacionadas más directamente con el software quizá sean las que han acogido con mayor frialdad la noticia debido a que opinan que tanto si aumenta como si disminuye la venta de Sinclair, el parque de software va a seguir siendo prácticamente el mismo, y creen que éste es un asunto que repercute más directamente en los vendedores de hardware. Sin embargo, también creen que sería muy lamentable que Amstrad dejara a los ordenadores Sinclair en segunda fila, pues el éxito y la popularidad que han alcanzado estos modelos es algo indudable.

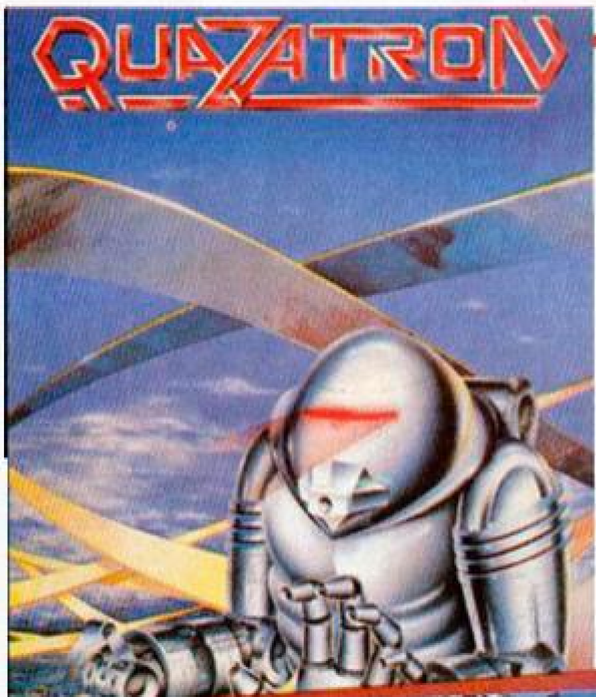
Por su parte, las casas dedicadas a comercializar estos productos de Sinclair se sienten ligeramente más alarmadas y esta medida ha causado reacciones discrepantes entre sí.

Las opiniones se dividen: por un lado hay quien piensa que esta nueva situación va a resultar completamente positiva debido a que se va a poner fin a la embarazosa situación económica que Sinclair estaba atravesando en los últimos tiempos y que Allan Sugar, director general de Amstrad, sabrá darles un gran impulso a los actuales modelos de Sinclair, así como a los posibles venideros.

Sin embargo, también hay quien piensa que esta nueva relación va a tener el aspecto negativo de que Amstrad se va a convertir en un auténtico monopolio que va a poder controlar a su antojo todo el mercado. También se piensa que la compañía Amstrad nunca se ha preocupado del usuario y que su único objetivo siempre ha sido el de vender más y más ordenadores, viendo toda la situación desde el punto de vista del negocio.

Como se ve, las opiniones son ligeramente distintas entre sí y esta nueva situación ha despertado reacciones muy diversas en cuanto al posible futuro de los ordenadores Sinclair y de sus repercusiones en el usuario.

A pesar de ello, se puede deducir que esta nueva relación comercial Amstrad —Sinclair supone una nueva chispa de esperanza para la ya casi apagada luz de Sinclair y que, aunque aún tendremos que dejar pasar algún tiempo para que los acontecimientos sigan su curso lógico, es de suponer que las cosas a nivel del usuario van a continuar prácticamente igual. Eso, a falta de una mejora considerable, sería lo mejor que podría ocurrir.



QUAZATRON: La novedad de Hewson

Quazatron representa la última incursión de Hewson Consultants en el mundo del software. Tras alcanzar la fama con juegos como Avalon, Dragontorc, Technician Ted o Suthern Belle, vuelven a acaparar la atención de los usuarios con esta creativa e interesante mezcla de arcade y estrategia.

En este programa, muy parecido en su concepción al famoso Gyroscope, nos situamos al mando de un robot rotatorio a quien se le ha asignado la misión de limpiar la escabrosa superficie de la ciudad de Quazatron de una molesta y antipática invasión alienígena.



Para destruir a estos androides disponemos de varias opciones: disparar contra ellos, programar las rutas para su autodestrucción o aprisionándoles.

Dependiendo de las circunstancias, deberemos utilizar unos u otros sistemas de destrucción, por lo que además de habilidad en el manejo de este KLP-2, necesitaremos tener a punto nuestras dotes de estratega para saber sacar el máximo partido a cada situación y poder actuar con más rapidez y precisión.

Evidentemente, no es el momento más oportuno para hacer una valoración y un comentario detallado a cerca de los pormenores del desarrollo de Quazatron y sus características más notables, pero sí podemos decir que, en general, es un programa que presenta una buena calidad gráfica y que tiene trazas de resultar bastante adictivo.

Su lanzamiento en España se espera para dentro de muy pocas semanas, aunque no podemos precisar con exactitud cuando se pondrá a la venta.

NUEVOS MONITORES HANTAREX

Tres nuevos modelos de monitores han sido presentados en el mercado por Erbatrón, distribuidor en España de los productos de HANTAREX.

Estos monitores, que pueden ser utilizados con los ordenadores Sinclair, son los siguientes: Boxer 12 PC, Hanty y CT 900.

Sus características más destacables son las siguientes:

Boxer 12 PC: alta resolución, monocromo, 12 pulgadas, conexión de vídeo, dimensiones: 309, 280, 264 mm, peso 7,5 kg.

Hanty: alta resolución, monocromo, 12 pulgadas, fósforo verde, señal de vídeo, dimensiones: 314, 285, 305 mm, peso 7,5 kg.

CT 900: color, resolución: 600 pixels o 312 líneas, 28 pulgadas, dimensiones: 635, 590, 450, peso 35 kg.

Los tres modelos pertenecen a las últimas novedades presentadas este año dentro de la gama de monitores HANTAREX y ya están disponibles en el mercado.



CREADA UNA ASOCIACION EUROPEA DE DISTRIBUIDORES

Recientemente se ha creado, con sede en Gran Bretaña, una asociación que reúne a algunas de las compañías distribuidoras de software más importantes de la Europa continental.

Esta asociación lleva el nombre de Micropool International Ltd. y tiene como principal objetivo el de impulsar y desarrollar más ampliamente el mercado del software inglés en el resto de los países europeos.

Micropool guarda una estrecha relación con un buen número de casas creadoras de software británico, quienes han ofrecido sus productos a todas estas distribuidoras.

La compañía está formada por miembros de nueve países diferentes: Alemania, Francia, España (con SERMA), Suecia, Noruega, Finlandia, Dinamarca, Italia y Bélgica.

Uno de los primeros productos que Micropool distribuirá en nuestro país será el Speedking Joystick de Konix, la casa que se hizo famosa por su Grip joystick, el primero con diseño anatómico, y parece ser que en un futuro próximo tiene pensado continuar con la distribución de otros periféricos para Spectrum.

20 +

CLASIFICACION	SEMANAS PERM.	TENDENCIA	20 +	SPECTRUM	AMSTRAD	COMMODORE	MSX
1			**RAMBO.** Ocean	●	●	●	
2			**CAMELOT WARRIORS.** Dinamic	●			
3			**SABOTEUR.** Durell	●	●		
4			**SIR FRED.** Made in Spain	●	●		
5			**LEYEND OF AMAZON WOMEN.** Silver Soft	●			
6			**MOVIE.** Imagine	●			
7			**YIE AR KUNG-FU.** Imagine	●	●	●	●
8			**COMANDO.** Elite	●		●	
9			**WORLD SERIES BASKETBALL.** Imagine	●			
10			**THE WAY OF THE EXPLODING FIST.** Melbourne House	●	●	●	
11			**MATCH DAY.** Artic	●	●		
12			**SUPERTEST.** Ocean	●	●		●
13			**THREE WEEKS IN PARADISE.** Mikro-Gen	●	●		
14			**POLE POSITION.** Atari	●	●		
15			**MICKIE.** Imagine	●		●	
16			**PROFANATION.** Dinamic	●			
17			**WINTER GAMES.** Epyx	●		●	
18			**TURBO ESPRIT.** Durell	●			
19			**OLE TORO.** Dinamic	●			
20			**PENTAC.** Compulogical	●			

Con la intención de ofreceros una lista que corresponda fielmente a la realidad del índice de ventas alcanzado por los distintos programas disponibles en el mercado, hemos decidido solicitar la colaboración de los departamentos de informática de los centros de El Corte Inglés de toda España.

En el número anterior os ofrecimos la lista de los juegos más vendidos para todos los ordenadores.

A partir de esta semana, todas las listas se referirán exclusivamente a las ventas alcanzadas para Spectrum, independientemente de que estén disponibles para otros ordenadores.

El Corte Inglés

EL TRIANGULO

F. GARCIA

Spectrum 48 K

TODAS LAS LETRAS MAYUSCULAS SUBRAYADAS DEBERAN TECLEARSE EN MODO GRAFICO.

Este juego va de comer, pero no empieces a relamerte que lo que hay que comer no es un suculento menú sino todas las fichas que componen el triángulo.

El tablero triangular, como puedes ver en la foto, está relleno de fichas que tendremos que ir comiendo, pero siempre, siguiendo estas reglas:

— En todos los movimientos es obligatorio comer fichas.

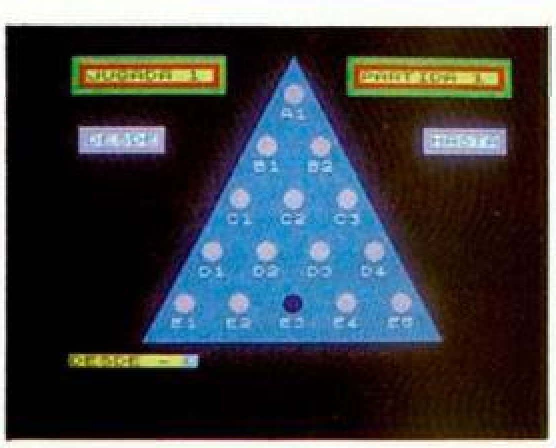
— Para comer hay que saltar tras la ficha que quieras y que debe ser la que se tenga más cerca, siempre que tras ella exista una posición libre.

— Lógicamente, si lo consigues, al final no quedará más que una ficha.

— Si tras 50 partidas no lo has conseguido, el ordenador te preguntará si quieres la solución, aunque ésta no es la única posible.

— Si ves que alguna partida ya no tiene solución, introduce "00", siempre que tengas como mínimo 7 movimientos efectuados.

```
  90 BORDER 0: PAPER 0: INK 7: C
LS
 100 GO SUB 9000
 112 LET part=1
 118 CLS : GO SUB 9100
 120 LET a1=1: LET b1=1: LET b2=
1: LET c1=1: LET c2=1: LET c3=1:
 LET d1=1: LET d2=1: LET d3=1: L
ET d4=1: LET e1=1: LET e2=1: LET
 e3=0: LET e4=1: LET e5=1
 125 LET mov=1: LET fich=14
 140 POKE 23658,8: INK 7
 190 LET gui=9
 200 INPUT PAPER 6; INK 0;"DESDE
 - "; LINE d$: IF d$<>"00" AND d
$<>"A1" AND d$<>"B1" AND d$<>"B2
" AND d$<>"C1" AND d$<>"C2" AND
d$<>"C3" AND d$<>"D1" AND d$<>"D
2" AND d$<>"D3" AND d$<>"D4" AND
 d$<>"E1" AND d$<>"E2" AND d$<>"
E3" AND d$<>"E4" AND d$<>"E5" TH
EN GO TO 200
 205 IF d$="00" AND mov>=8 THEN
FOR f=9 TO gui: PRINT AT f,2;"
";AT f,28;"  ": NEXT f: GO SUB 9
200: LET part=part+1: PRINT PAPE
R 6; INK 0;AT 1,8;"1 ";AT 1,29;p
art: GO TO 120
 210 BEEP .05,10: PRINT AT gui,2
;d$
 220 INPUT PAPER 6; INK 0;"HASTA
 - "; LINE h$: IF h$<>"00" AND h
$<>"A1" AND h$<>"B1" AND h$<>"B2
" AND h$<>"C1" AND h$<>"C2" AND
h$<>"C3" AND h$<>"D1" AND h$<>"D
2" AND h$<>"D3" AND h$<>"D4" AND
 h$<>"E1" AND h$<>"E2" AND h$<>"
E3" AND h$<>"E4" AND h$<>"E5" TH
EN GO TO 220
 225 IF h$="00" AND mov>=8 THEN
FOR f=9 TO gui: PRINT AT f,2;"
";AT f,28;"  ": NEXT f: GO SUB 9
200: LET part=part+1: PRINT PAPE
R 6; INK 0;AT 1,8;"1 ";AT 1,29;p
art: GO TO 120
 230 BEEP .05,10: PRINT AT gui,2
8;h$
 240 IF (d$="A1" AND h$="C1" AND
 a1=1 AND b1=1 AND c1=0) OR (d$=
"A1" AND h$="C3" AND a1=1 AND b2
=1 AND c3=0) OR (d$="B1" AND h$=
"D1" AND b1=1 AND c1=1 AND d1=0)
 OR (d$="B1" AND h$="D3" AND b1=
1 AND c2=1 AND d3=0) THEN GO TO
1100
 245 IF (d$="B2" AND h$="D2" AND
 b2=1 AND c2=1 AND d2=0) OR (d$=
"B2" AND h$="D4" AND b2=1 AND c3
=1 AND d4=0) OR (d$="C1" AND h$=
"E1" AND c1=1 AND d1=1 AND e1=0)
 OR (d$="C1" AND h$="E3" AND c1=
1 AND d2=1 AND e3=0) THEN GO TO
1100
 250 IF (d$="C2" AND h$="E2" AND
 c2=1 AND d2=1 AND e2=0) OR (d$=
"C2" AND h$="E4" AND c2=1 AND d3
=1 AND e4=0) OR (d$="C3" AND h$=
"E3" AND c3=1 AND d3=1 AND e3=0)
 OR (d$="C3" AND h$="E5" AND c3=
1 AND d4=1 AND e5=0) THEN GO TO
1100
 260 IF (d$="C1" AND h$="A1" AND
 a1=0 AND b1=1 AND c1=1) OR (d$=
"C3" AND h$="A1" AND a1=0 AND b2
=1 AND c3=1) OR (d$="D1" AND h$=
"B1" AND b1=0 AND c1=1 AND d1=1)
 OR (d$="D3" AND h$="B1" AND b1=
0 AND c2=1 AND d3=1) THEN GO TO
1200
 265 IF (d$="D2" AND h$="B2" AND
 b2=0 AND c2=1 AND d2=1) OR (d$=
"D4" AND h$="B2" AND b2=0 AND c3
=1 AND d4=1) OR (d$="E1" AND h$=
"C1" AND c1=0 AND d1=1 AND e1=1)
 OR (d$="E3" AND h$="C1" AND c1=
0 AND d2=1 AND e3=1) THEN GO TO
1200
 270 IF (d$="E2" AND h$="C2" AND
 c2=0 AND d2=1 AND e2=1) OR (d$=
"E4" AND h$="C2" AND c2=0 AND d3
=1 AND e4=1) OR (d$="E3" AND h$=
"C3" AND c3=0 AND d3=1 AND e3=1)
 OR (d$="E5" AND h$="C3" AND c3=
0 AND d4=1 AND e5=1) THEN GO TO
1200
 280 IF (d$="C1" AND h$="C3" AND
 c1=1 AND c2=1 AND c3=0) OR (d$=
"D1" AND h$="D3" AND d1=1 AND d2
=1 AND d3=0) OR (d$="D2" AND h$=
"D4" AND d2=1 AND d3=1 AND d4=0)
 THEN GO TO 1300
 285 IF (d$="E1" AND h$="E3" AND
 e1=1 AND e2=1 AND e3=0) OR (d$=
"E2" AND h$="E4" AND e2=1 AND e3
=1 AND e4=0) OR (d$="E3" AND h$=
"E5" AND e3=1 AND e4=1 AND e5=0)
 THEN GO TO 1300
 290 IF (d$="C3" AND h$="C1" AND
 c1=0 AND c2=1 AND c3=1) OR (d$=
"D3" AND h$="D1" AND d1=0 AND d2
=1 AND d3=1) OR (d$="D4" AND h$=
"D2" AND d2=0 AND d3=1 AND d4=1)
 THEN GO TO 1400
 295 IF (d$="E3" AND h$="E1" AND
 e1=0 AND e2=1 AND e3=1) OR (d$=
"E4" AND h$="E2" AND e2=0 AND e3
=1 AND e4=1) OR (d$="E5" AND h$=
"E3" AND e3=0 AND e4=1 AND e5=1)
 THEN GO TO 1400
1000 PRINT PAPER 5; INK 0;AT 4,0
;"MOVIMIENTO";AT 4,22;"INCORRECT
O"
1020 FOR f=0 TO 20: BEEP .05,50:
 NEXT f: PRINT AT 4,0;"
";AT 4,22;"
```



```
1030 PRINT AT gui,2;"  ";AT gui,
28;"  ": GO TO 200
1101 IF d$="A1" AND h$="C1" THEN
 LET a1=0: LET b1=0: LET c1=1
1102 IF d$="A1" AND h$="C3" THEN
 LET a1=0: LET b2=0: LET c3=1
1104 IF d$="B1" AND h$="D1" THEN

 LET b1=0: LET c1=0: LET d1=1
1106 IF d$="B1" AND h$="D3" THEN
 LET b1=0: LET c2=0: LET d3=1
1108 IF d$="B2" AND h$="D2" THEN
 LET b2=0: LET c2=0: LET d2=1
1110 IF d$="B2" AND h$="D4" THEN
 LET b2=0: LET c3=0: LET d4=1
1112 IF d$="C1" AND h$="E1" THEN
 LET c1=0: LET d1=0: LET e1=1
1114 IF d$="C1" AND h$="E3" THEN
 LET c1=0: LET d2=0: LET e3=1
1116 IF d$="C2" AND h$="E2" THEN
 LET c2=0: LET d2=0: LET e2=1
1116 IF d$="C2" AND h$="E4" THEN
 LET c2=0: LET d3=0: LET e4=1
1120 IF d$="C3" AND h$="E3" THEN
 LET c3=0: LET d3=0: LET e3=1
1122 IF d$="C3" AND h$="E5" THEN
 LET c3=0: LET d4=0: LET e5=1
1201 IF d$="C1" AND h$="A1" THEN
 LET a1=1: LET b1=0: LET c1=0
1202 IF d$="C3" AND h$="A1" THEN

 LET a1=1: LET b2=0: LET c3=0
1204 IF d$="D1" AND h$="B1" THEN
 LET b1=1: LET c1=0: LET d1=0
1206 IF d$="D3" AND h$="B1" THEN
 LET b1=1: LET c2=0: LET d3=0
1208 IF d$="D2" AND h$="B2" THEN
 LET b2=1: LET c2=0: LET d2=0
1210 IF d$="D4" AND h$="B2" THEN
 LET b2=1: LET c3=0: LET d4=0
1212 IF d$="E1" AND h$="C1" THEN
 LET c1=1: LET d1=0: LET e1=0
1214 IF d$="E3" AND h$="C1" THEN
 LET c1=1: LET d2=0: LET e3=0
1216 IF d$="E2" AND h$="C2" THEN
 LET c2=1: LET d2=0: LET e2=0
1218 IF d$="E4" AND h$="C2" THEN
 LET c2=1: LET d3=0: LET e4=0
1220 IF d$="E3" AND h$="C3" THEN
 LET c3=1: LET d3=0: LET e3=0
1222 IF d$="E5" AND h$="C3" THEN
 LET c3=1: LET d4=0: LET e5=0
1301 IF d$="C1" AND h$="C3" THEN

 LET c1=0: LET c2=0: LET c3=1
1302 IF d$="D1" AND h$="D3" THEN
 LET d1=0: LET d2=0: LET d3=1
1304 IF d$="D2" AND h$="D4" THEN
 LET d2=0: LET d3=0: LET d4=1
1306 IF d$="E1" AND h$="E3" THEN
 LET e1=0: LET e2=0: LET e3=1
1308 IF d$="E2" AND h$="E4" THEN
 LET e2=0: LET e3=0: LET e4=1
1310 IF d$="E3" AND h$="E5" THEN
 LET e3=0: LET e4=0: LET e5=1
1401 IF d$="C3" AND h$="C1" THEN
 LET c1=1: LET c2=0: LET c3=0
1402 IF d$="D3" AND h$="D1" THEN
 LET d1=1: LET d2=0: LET d3=0
1404 IF d$="D4" AND h$="D2" THEN
 LET d2=1: LET d3=0: LET d4=0
1406 IF d$="E3" AND h$="E1" THEN
 LET e1=1: LET e2=0: LET e3=0
1408 IF d$="E4" AND h$="E2" THEN
 LET e2=1: LET e3=0: LET e4=0
1410 IF d$="E5" AND h$="E3" THEN

 LET e3=1: LET e4=0: LET e5=0
1600 LET fo=CODE d$(1)-51*(d$(1)
="E")-54*(d$(1)="D")-57*(d$(1)="
C")-60*(d$(1)="B")-63*(d$(1)="A"
)
1605 LET co=7+2*(d$="D1")+4*(d$=
"C1" OR d$="E2")+6*(d$="D2" OR d
$="B1")+8*(d$="E3" OR d$="C2" OR
 d$="A1")+10*(d$="D3" OR d$="B2"
)+12*(d$="E4" OR d$="C3")+14*(d$
="D4")+16*(d$="E5"): IF d$="E1"
THEN LET co=7
1610 LET fd=fo+8*(d$(1)<h$(1))-8
*(d$(1)>h$(1)): IF d$(1)=h$(1) T
HEN LET fd=fo
1615 LET cd=7+2*(h$="D1")+4*(h$=
"C1" OR h$="E2")+6*(h$="D2" OR h
$="B1")+8*(h$="E3" OR h$="C2" OR
 h$="A1")+10*(h$="D3" OR h$="B2"
)+12*(h$="E4" OR h$="C3")+14*(h$
="D4")+16*(h$="E5"): IF h$="E1"
THEN LET cd=7

1620 LET fi=fo+4*(d$(1)<h$(1))-4
*(d$(1)>h$(1)): IF fo=fd THEN LE
T fi=fo
1625 LET ci=co-2*(d$(1)<>h$(1) A
ND d$(2)>=h$(2) AND co>cd)+2*(d$
(1)<>h$(1) AND d$(2)<=h$(2) AND
co<cd)+4*(d$(1)=h$(1) AND d$(2)<
h$(2))-4*(d$(1)=h$(1) AND d$(2)>
h$(2))
2000 FOR f=1 TO 10: PRINT INK 3;
 PAPER 1;AT fo,co;"AB";AT fo+1,c
o;"CD": BEEP .005,30: PRINT INK
0; PAPER 1;AT fo,co;"AB";AT fo+1
,co;"CD": NEXT f: PRINT PAPER 1;
 INK 0;AT fo,co;"AB";AT fo+1,co;
"CD"
2010 FOR f=1 TO 10: PRINT INK 3;
 PAPER 1;AT fd,cd;"AB";AT fd+1,c
d;"CD": BEEP .005,50: PRINT INK
0; PAPER 1;AT fd,cd;"AB";AT fd+1
,cd;"CD": NEXT f: PRINT PAPER 1;
 INK 3;AT fd,cd;"AB";AT fd+1,cd;

"CD"
2020 PRINT INK 0; PAPER 1;AT fi,
ci;"AB";AT fi+1,ci;"CD"
2999 LET mov=mov+1: PRINT PAPER
6; INK 0;AT 1,8;mov: LET gui=gui
+1
3000 LET fich=fich-1: IF fich=1
THEN GO TO 3025
3020 GO TO 200
3025 PRINT INK 6; PAPER 0; FLASH
 1;AT 4,0;"!!CORRECTO!";AT 4,20;
"!!CORRECTO!!"
3030 POKE 50007,7: FOR j=1 TO 6:
 FOR f=10 TO 25: OUT 254,f: POKE
 50015,f: POKE 50001,f: RANDOMIZ
E USR 50000: NEXT f: NEXT j
3040 INPUT PAPER 6; INK 2;"TE AT
REVES OTRA VEZ ? (S/N) "; LINE r
$: IF r$="S" THEN CLS : LET part
=part+1: GO TO 115
3050 RUN
6700 PRINT PAPER 1; INK 6; FLASH
 1;AT 4,0;"PON ATENCION";AT 4,20
;"PON ATENCION"
6750 FOR f=1 TO 300: NEXT f
6800 LET sol=7050
6900 RESTORE sol: READ fo,co,fd,
cd,fi,ci
7000 FOR f=1 TO 10: PRINT INK 3;
 PAPER 1;AT fo,co;"AB";AT fo+1,c
o;"CD": BEEP .005,30: PRINT INK
0; PAPER 1;AT fo,co;"AB";AT fo+1
,co;"CD": NEXT f: PRINT PAPER 1;
 INK 0;AT fo,co;"AB";AT fo+1,co;
"CD"
7010 FOR f=1 TO 10: PRINT INK 3;
 PAPER 1;AT fd,cd;"AB";AT fd+1,c
d;"CD": BEEP .005,50: PRINT INK
0; PAPER 1;AT fd,cd;"AB";AT fd+1
,cd;"CD": NEXT f: PRINT PAPER 1;
 INK 3;AT fd,cd;"AB";AT fd+1,cd;
"CD"
7020 PRINT INK 0; PAPER 1;AT fi,
ci;"AB";AT fi+1,ci;"CD"

7040 LET sol=sol+5: IF sol<7115
THEN GO TO 6900
7045 PRINT PAPER 2; INK 6; FLASH
 1;AT 4,0;"ESTA CLARO ?";AT 4,20
;"PUES HALA!!!"
7046 FOR f=1 TO 300: NEXT f
7047 PRINT AT 4,0;"            "
;AT 4,20;"            ": RETURN
7050 DATA 10,19,18,15,14,17
7055 DATA 2,15,10,19,6,17
7060 DATA 6,13,14,17,10,15
7065 DATA 14,21,6,17,10,19
7070 DATA 14,9,6,13,10,11
7075 DATA 18,11,10,15,14,13
7080 DATA 18,15,10,19,14,17
7085 DATA 6,17,14,21,10,19
7090 DATA 6,13,14,17,10,15
7095 DATA 18,23,10,19,14,21
7100 DATA 10,19,18,15,14,17
7105 DATA 18,19,18,11,18,15
7110 DATA 18,7,18,15,18,11
9000 RESTORE 9010: FOR f=USR "a"
 TO USR "h"+7: READ n: POKE f,n:
 NEXT f
9005 RESTORE 9030: FOR f=50000 T
O 50024: READ n: POKE f,n: NEXT
f
9006 RETURN
9010 DATA 0,0,3,15,31,31,63,63,0
,0,192,240,248,248,252,252,63,63
,31,31,15,3,0,0,252,252,248,248,
240,192,0,0
9020 DATA 1,1,3,3,7,7,15,15,31,3
1,63,63,127,127,255,255,128,128,
192,192,224,224,240,240,248,248,
252,252,254,254,255,255
9030 DATA 17,1,0,33,10,0,6,10,19
7,213,229,205,181,3,1,1,0,225,23
7,74,209,193,16,240,201
9100 PRINT INK 4; PAPER 2;AT 0,0
;"          ";AT 0,20;"          
  ";AT 1,0;" ";AT 1,10;" ";AT
1,20;" ";AT 1,31;" ";AT 2,0;"
         ";AT 2,20;"           "

9105 PRINT PAPER 6; INK 0;AT 1,1
;"JUGADA 1 ";AT 1,21;"PARTIDA 1
"

9200 RESTORE 9210: LET c=15: FOR
 f=0 TO 21: READ x$: PRINT INK 1
;AT f,c;x$: LET c=c-.5: NEXT f
9210 DATA "EG","FH","E G","F H
","E  G","F  H","E   G","
F   H","E    G","F    H
","E     G","F     H","E      G","F      
 H"
9215 DATA "E       G","F
       H","E        
G","F        H","E         G","F         H","E          G"
,"F          H"
9500 LET a$="AB  AB  AB  AB  AB"
: LET b$="CD  CD  CD  CD  CD"
9505 LET a=15: LET b=2
9510 FOR f=2 TO 18 STEP 4: PRINT
  INK 3; PAPER 1;AT f,a;a$( TO b
);AT f+1,a;b$( TO b): LET a=a-2:
 LET b=b+4: NEXT f
9512 PRINT INK 0; PAPER 1;AT 18,
15;"AB";AT 19,15;"CD"
9515 LET a=15: LET b=4
9520 FOR f=1 TO 5: READ l$: PRIN
T INK 7; PAPER 1;AT b,a;l$: LET
a=a-2: LET b=b+4: NEXT f
9525 DATA "A1","B1  B2","C1  C2
 C3","D1  D2  D3  D4","E1  E2  E
3  E4  E5"
9532 PRINT PAPER 6; INK 0;AT 1,2
9;part
9534 PRINT INK 3;AT 5,0;"       
";AT 5,25;"       ";AT 6,0;" ";A
T 6,6;" ";AT 6,25;" ";AT 6,31;"
";AT 7,0;"       ";AT 7,25;"
  "

9535 PRINT INK 1; PAPER 7;AT 6,1
;"DESDE";AT 6,26;"HASTA"
9540 IF part>=50 THEN BEEP .5,10
: INPUT PAPER 4; INK 0;"QUIERES
UNA SOLUCION ? (S/N) "; LINE r$:
 IF r$="S" THEN GO SUB 6700: GO
TO 9200
9545 RETURN
```

TRUCOS

LLIST n TO m

Cuando necesitamos obtener el listado por impresora de una parte de programa, no en su totalidad, echamos en falta en el Spectrum una sentencia como ésta: LLIST n TO m. Es decir, una sentencia que imprimiera por impresora desde línea n a la línea m, ambas inclusive.

Pues bien, el programa siguiente realiza esta tarea:

```
9990 REM  LLIST n TO m
9991 INPUT "Linea inicial? ";li:
 INPUT "Linea final? ";lf
9992 LET d=PEEK 23635+256*PEEK 2
3636
9993 LET n=256*PEEK d+PEEK (d+1)
: LET l=PEEK (d+2)+256*PEEK (d+3
)
9994 IF n<li THEN LET d=d+l+4: G
O TO 9993
9995 IF n>lf THEN STOP
9996 LET A$="    "+STR$ n+" ": L
ET A$=A$(LEN A$-4 TO ): LET d=d+
4
9997 IF PEEK d=14 THEN LET d=d+6
9998 IF PEEK d=13 THEN LET d=d+1
: LPRINT A$: GO TO 9993
9999 LET A$=A$+CHR$ PEEK d: LET
d=d+1: GO TO 9997
```

Para utilizarlo:

1) Cargar el programa que queremos listar por fragmentos.

2) Cargar el programa «LLIST n TO m» con MERGE "".

3) Hacer UN GO TO 9990.

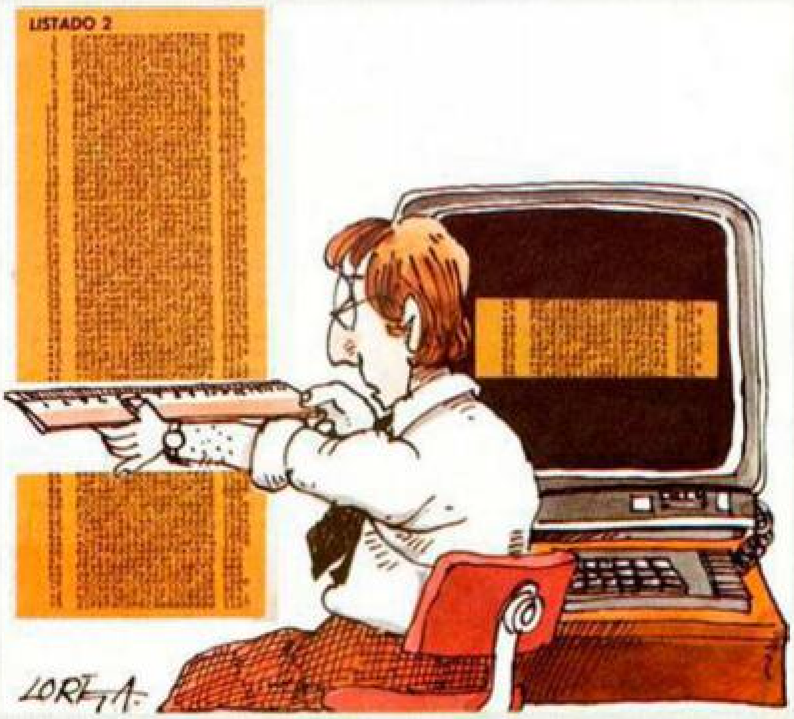
4) Introducir el número de líneas a partir del cual empezará la impresión del listado.

5) Introducir el último número de línea del fragmento de programa.

6) El programa «LLIST n TO m» hará todo lo demás.

7) Si nuestra impresora es del tipo ZX-Pinter, cambiaremos en la línea 9996 LEN A$ —3, en lugar de —4.

Este truco que es más una utilidad, se lo debemos a José M. Carmona.



UN «OUT» PARA EL BORDER

Manuel Marín Ardila ha conseguido un bonito efecto sobre el Border, utilizando para ello la instrucción OUT en una rutina en Código Máquina.

El listado en Assembler es el siguiente:

```
10        ORG  60000
20        LD   B,255
30 PRIM   LD   A,255
40 SEGU   OUT  (254),A
50        DEC  A
60        JR   NZ,SEGU
70        DEC  B
80        JR   NZ,PRIM
90        LD   A,7
100       OUT  (254),A
110       RET
```

Como véis es totalmente reubicable y si lo preferís, también os ofrecemos un pequeño cargador en Basic.

```
10 CLEAR 59999
20 FOR n=60000 TO 60016
30 READ a: POKE n,a
40 NEXT n
50 RANDOMIZE USR 60000
60 DATA 6,255,62,255,211,254,6
1,32,251,5,32,246,62,7,211,254,2
01
```

AUTODESTRUCCION

José Luis Chaves ha enviado unos pokes para conseguir la autodestrucción de un programa al intentar cortar los INPUTs tanto con STOP como usando CAPS SHIFT y 6 simultáneamente o al utilizar BREAK.

```
10 POKE 23613,2
20 POKE 23614,91
30 POKE 23298,0
40 POKE 23299,0
50 INPUT a$
60 GO TO 10
```

ENGAÑAR AL ORDENADOR?

No os resulta increíble la forma de trabajar de vuestro ordenador? Pues para que comprobeis que también puede fallar, intentar engañarle introduciendo un informe erróneo:

```
10 LET A$ = «VAL A$»
20 PRINT VAL A$
```

Una vez lanzado este programa con RUN queda la pantalla en blanco mientras el ordenador trata de calcular el resultado, dando al final con el error:

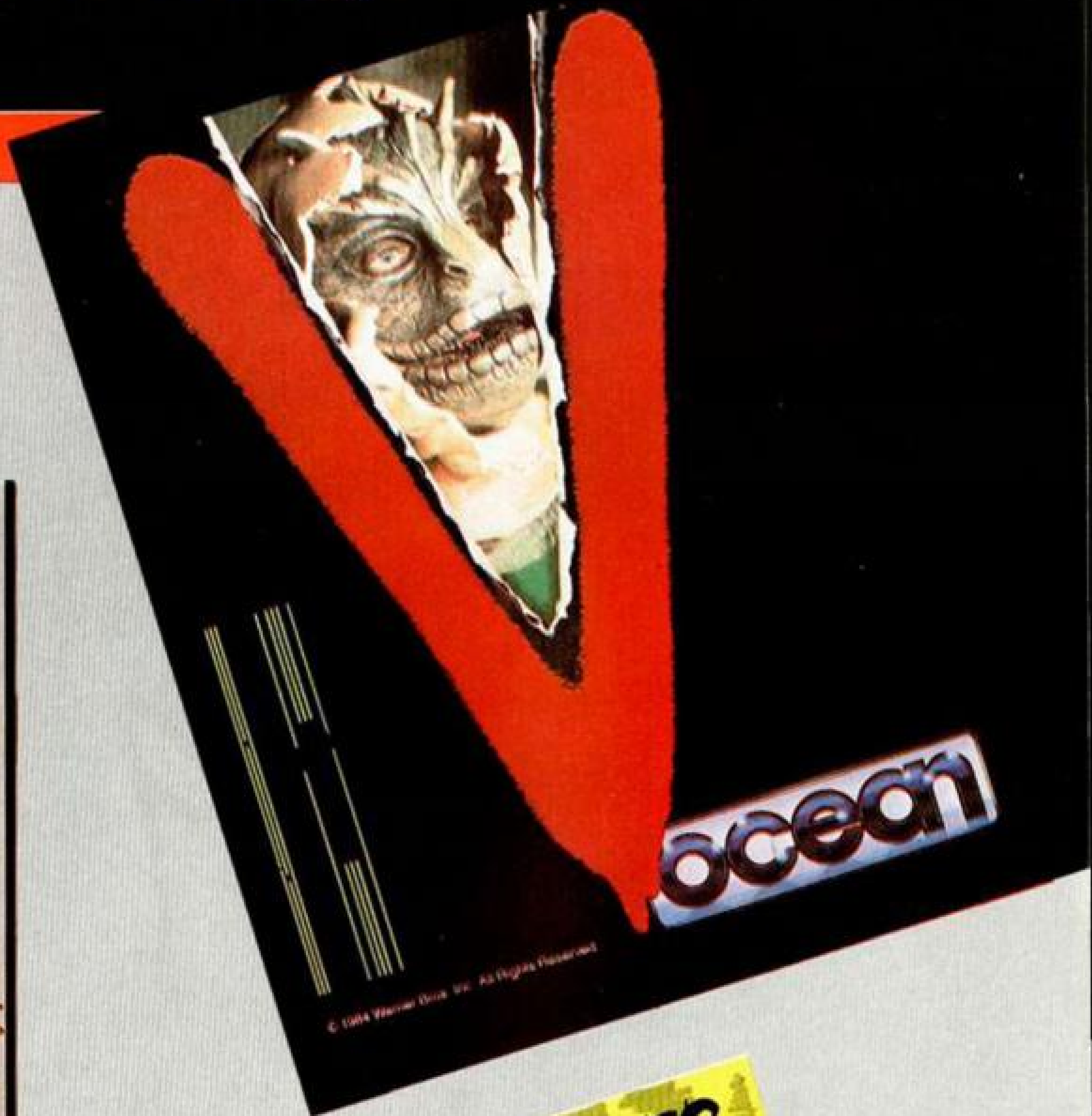
4 out of memory 20 : 1
y se para.

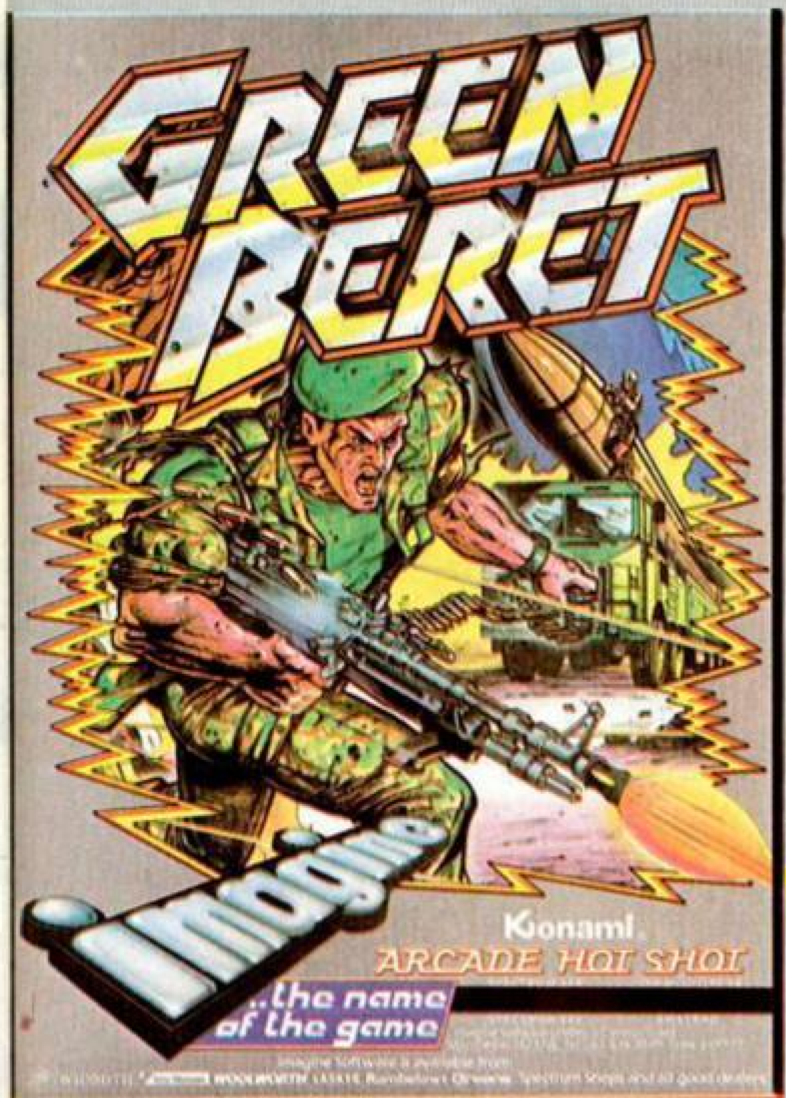
El problema reside en que el ordenador para hallar el valor de la expresión VAL A$ sustituye A$ por su valor (VAL A$) con los que tiene que recomenzar el proceso, llegando a consumirle toda la memoria.

ERBEdades

"V"

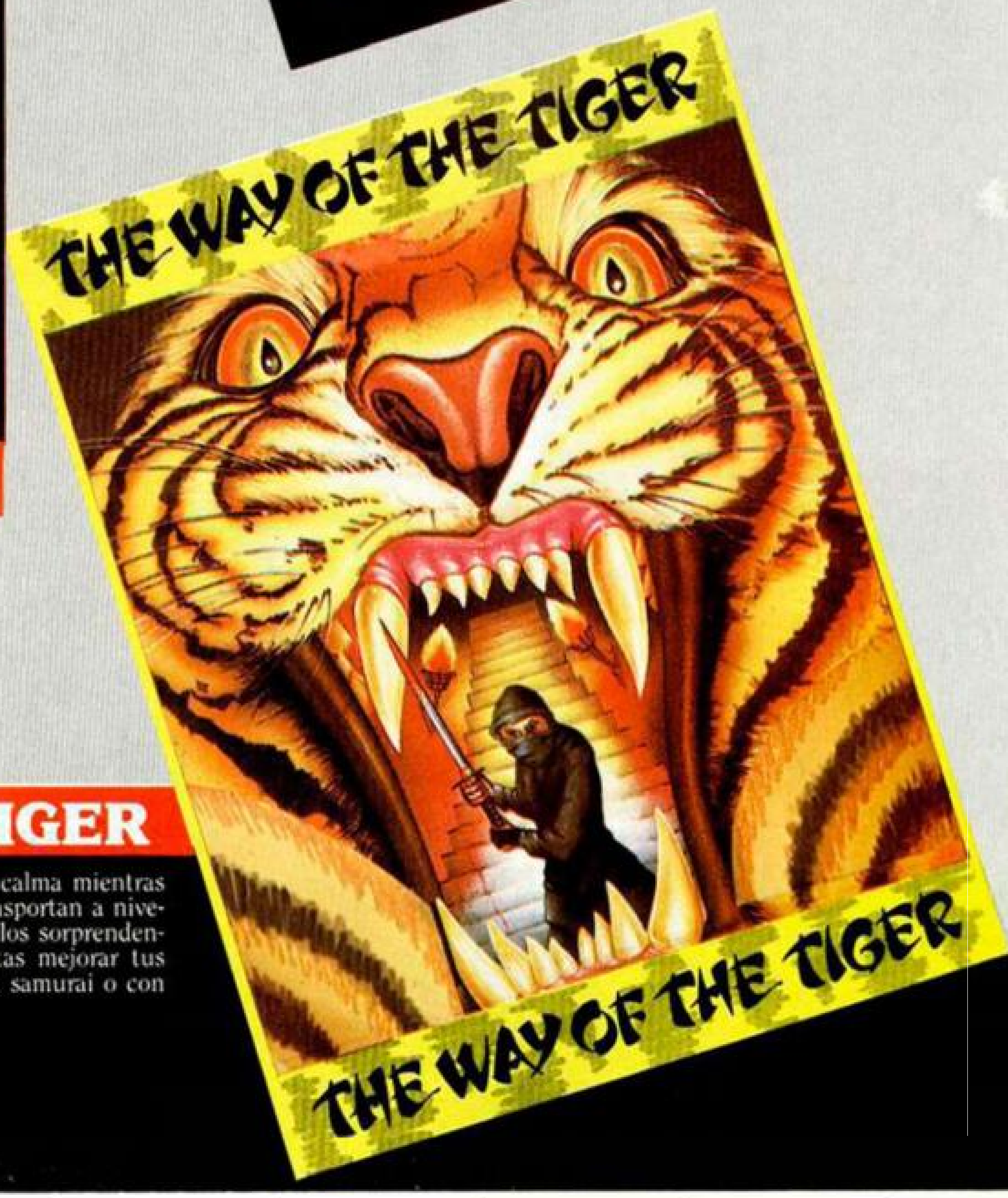
La serie de ciencia-ficción televisiva invade tu ordenador. La Tierra es visitada por alienígenas de aspecto humano. Si embargo, estos reptiles pronto revelarán su objetivo final... conquistar la Tierra y esclavizar a la raza humana.





GREEN BERET

Green Beret (boina verde), es el programa de mayor acción que hayas visto jamás en un ordenador.
Decir Green Beret es decir un hombre perfectamente entrenado para la lucha. Prepárate...



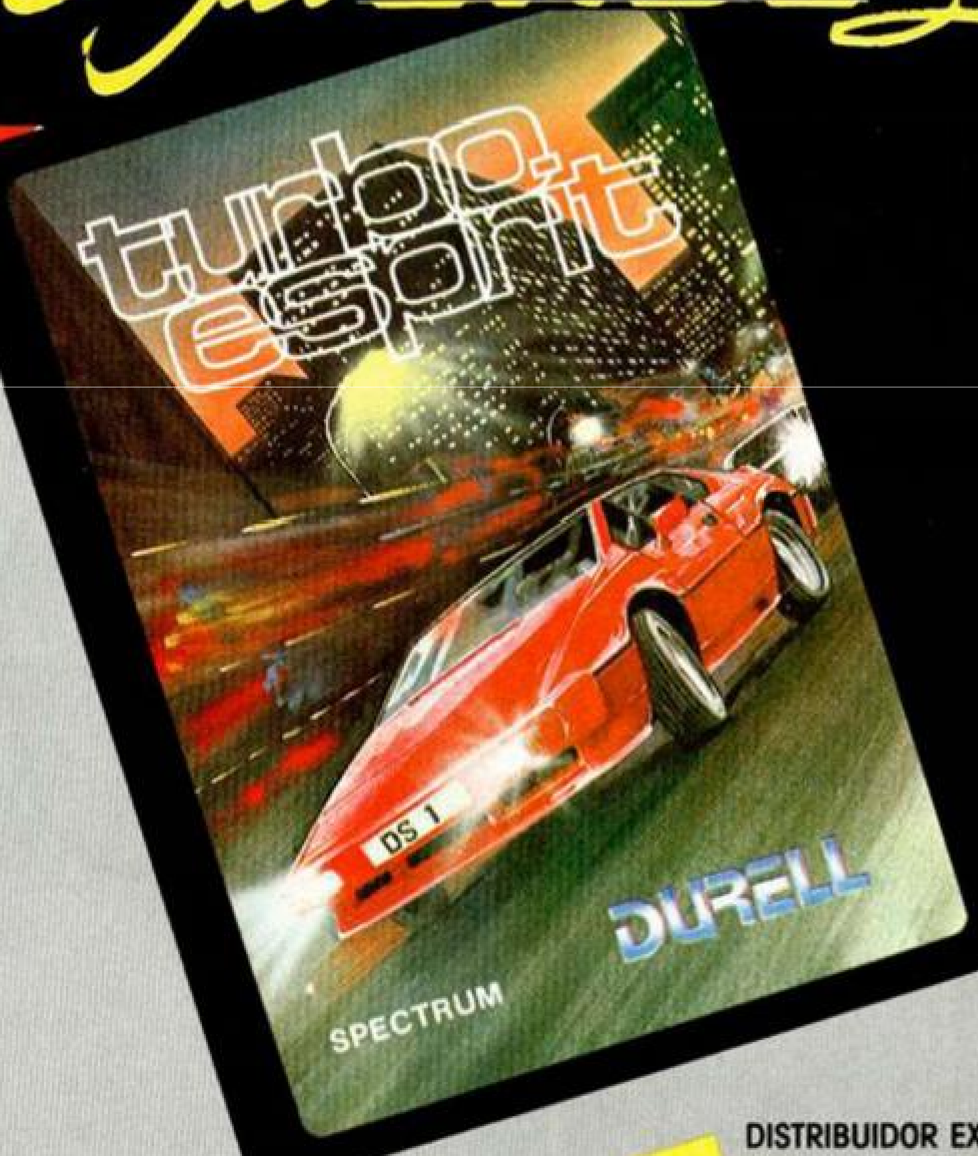
THE WAY OF THE TIGER

Entra en el mudo de los samurais. Mantén la calma mientras el movimiento y las rutinas de combate te transportan a niveles que nunca pensaste posibles. Experimenta los sorprendentes efectos del "Triple Scroll" mientras intentas mejorar tus técnicas de lucha cuerpo a cuerpo, con espada samurai o con mil posibilidades más.

Para que juERBEs

TURBO ESPRIT

Tu misión: vigilar y cuidar el cumplimiento de la ley que se ve amenazada por una terrible banda de delincuentes que han hecho del tráfico de narcóticos su negocio más rentable.
Tus medios: un Lotus Turbo Sprit dotado de uno de los máximos adelantos técnicos y con el que deberás patrullar por calles y avenidas.



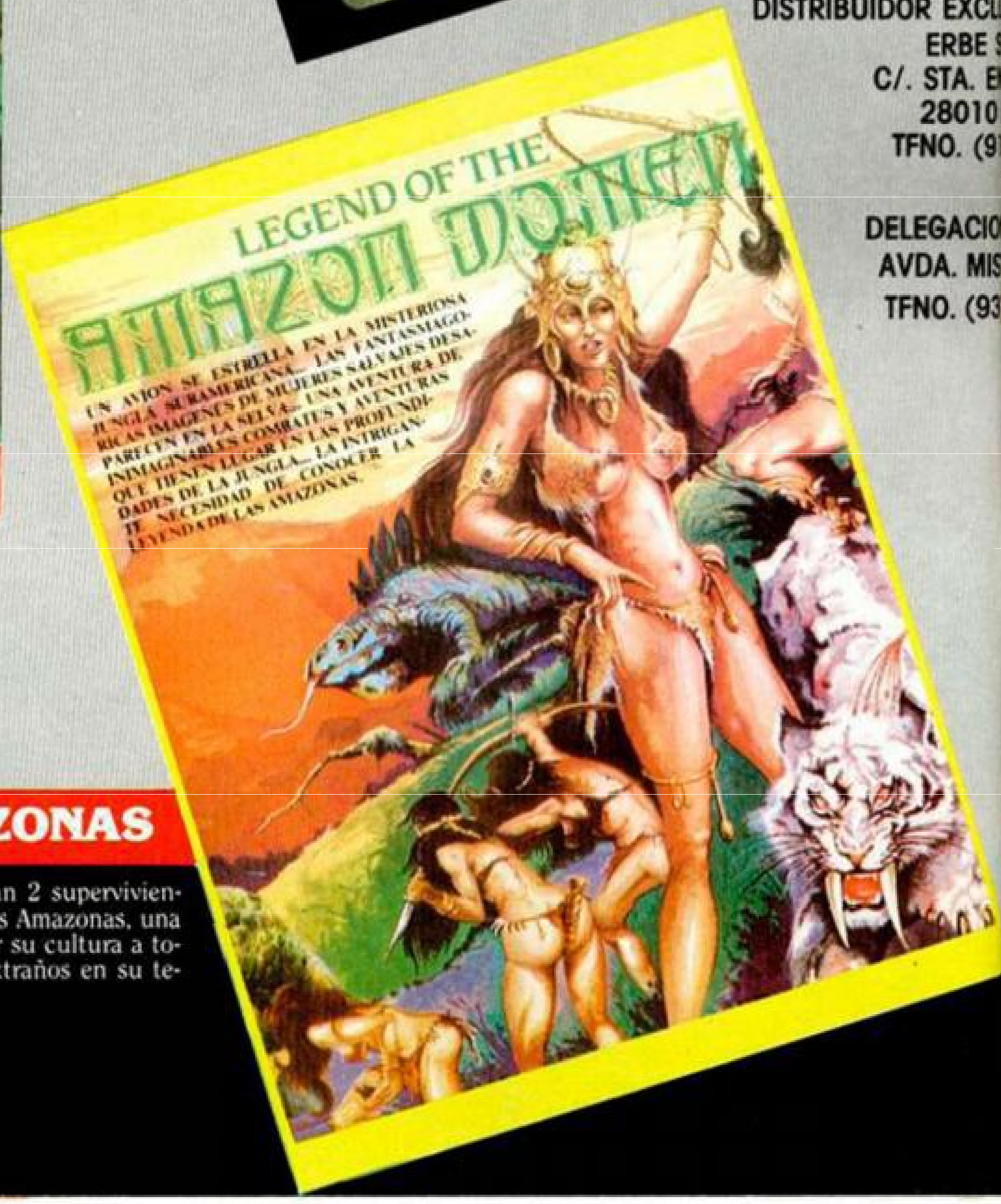


CYBERUN

Prepárate para una carrera sin límites a través del espacio con este nuevo juego de Ultimate. Los impresionantes gráficos y el inimitable estilo al que esta compañía nos tiene acostumbrados te transportarán a una auténtica aventura espacial a bordo del Cyberun.

DISTRIBUIDOR EXCLUSI
ERBE S
C/. STA. E
28010 M
TFNO. (91) 4

DELEGACION B
AVDA. MIS
TFNO. (93) 4



LA LEYENDA DE LAS AMAZONAS

Un avión se estrella en la jungla. Sólo quedan 2 supervivientes: lady Wilde y su hija que es raptada por las Amazonas, una raza de mujeres míticas que quieren conservar su cultura a toda costa y que no admiten la presencia de extraños en su territorio.

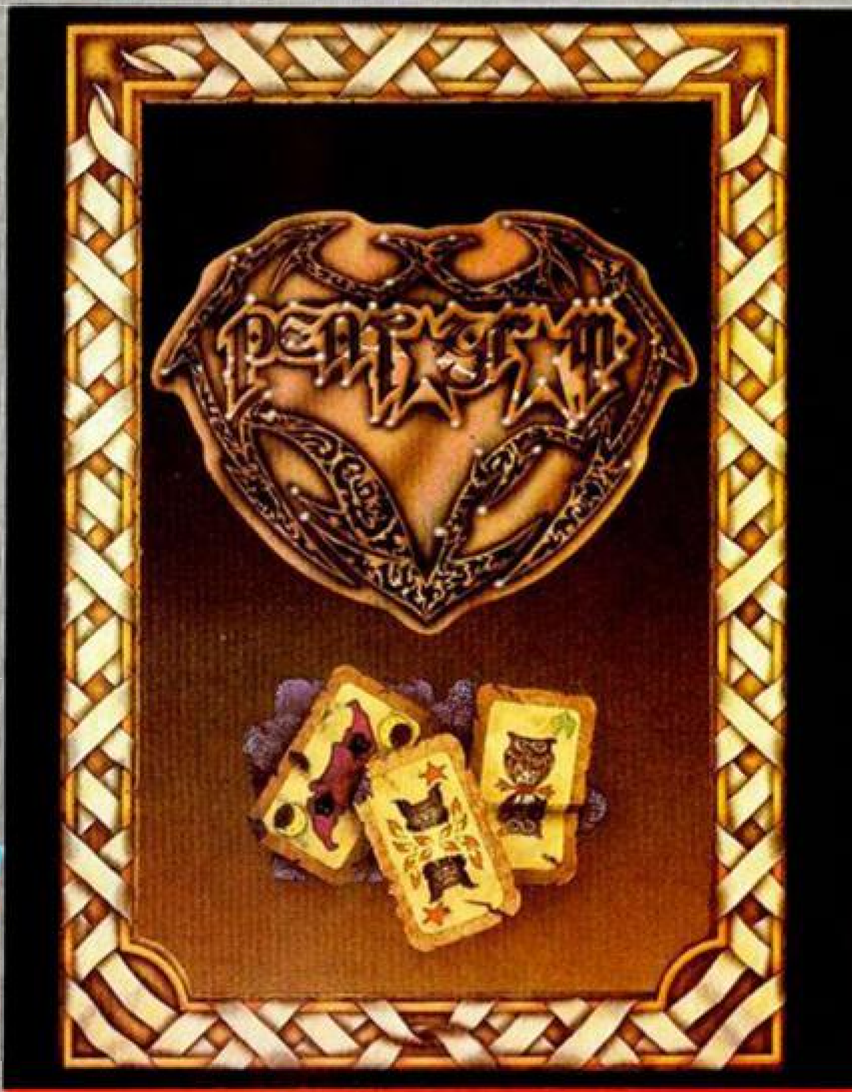


PING-PONG

La gran sorpresa. Gráficos increíbles, movimiento super-rápido, podrás efectuar las mismas jugadas que si tuvieras la paleta en tu mano.
Botes. rebotes, efectos, dejadas, saques, cortadas, mates... todo es posible con esta maravilla llamada "Ping-Pong".

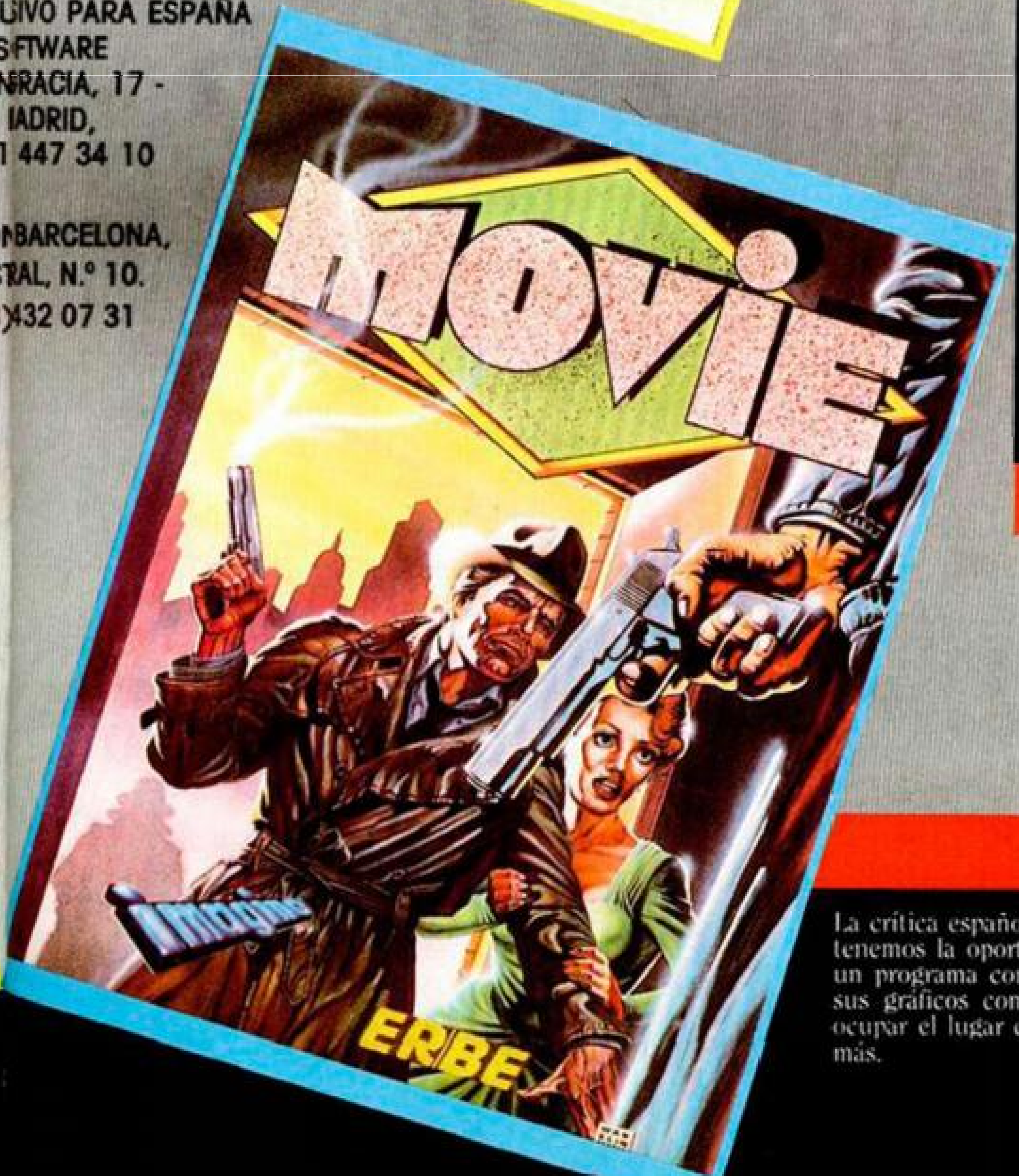
LUIVO PARA ESPAÑA
SFTWARE
ENRACIA, 17 -
IADRID,
1 447 34 10

ONBARCELONA,
STAL, N.º 10.
3)432 07 31



PENTAGRAM

El juego sorpresa de Ultimate que debes descubrir tú mismo.
... La continuación de "Saber Wulf".



MOVIE

La crítica española ha dicho de este juego: "Todos los días no tenemos la oportunidad de disfrutar en nuestro ordenador de un programa como éste. "Movie" es un juego que, tanto por sus gráficos como por su gran originalidad, está llamado a ocupar el lugar de una auténtica estrella." No hace falta decir más.

CYBERUN • Arcade • Ultimate

A LA BUSQUEDA DE LA CYBERNITA

Ultimate ha dado un giro de noventa grados a su tradicional método de programación y ha creado este Cyberun, un adictivo arcade de gran calidad y considerables dimensiones.

Ultimamente venía siguiendo una línea muy definida de creación. Desde que lanzara al mercado aquellas sus dos primeras maravillas, Knight Lore y Alien 8, siempre ha seguido trabajando en un estilo si no idéntico, sí similar de programación.



Sin embargo, Cyberun no tiene nada que ver con lo que había hecho hasta el momento.

Se trata de un arcade en toda la regla formado por innumerables pantallas, una nave y miles de objetos enemigos. El objetivo del juego consiste en recoger una serie de cristales, que en realidad corresponden al estado sólido de la cybernita (la principal fuente de energía del Universo). Estas porciones se encuentran repartidas por cinco planetas diferentes, y el conseguir recuperar todas nos va a resultar poco menos que imposible.

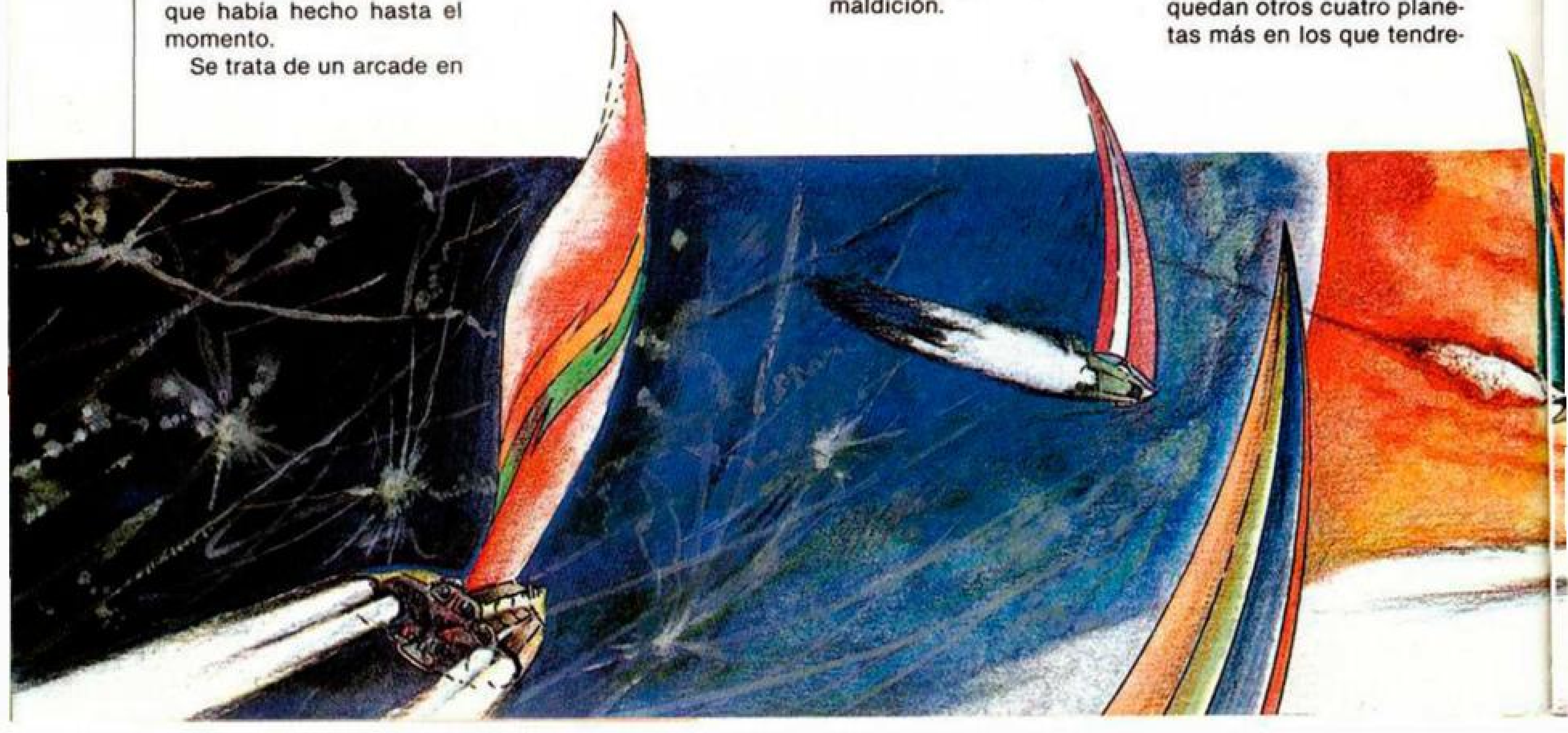
Pero antes de estar en disposición de recoger la cybernita, tendremos que llevar a cabo la tarea de composición de la nave. Esta se encuentra dividida en ocho fragmentos diferentes que se encuentran repartidos por la superficie del planeta. Estas piezas son las que nos permitirán llevar a cabo nuestra misión con éxito, pues son las que le imprimen a la nave una mejor maniobrabilidad, unas mayores posibilidades defensivas y también serán las que permitan arrancar los cristales incrustados en la dura roca.

Todo esto se consigue con impulsores láseres, cohetes, pinzas, etc..., y el llegar a completar correctamente la nave ya nos va a costar más de una desesperación y alguna que otra maldición.

Esto, como podréis comprobar, nos recuerda en cierta manera el argumento del Jet Pac, pero en esta ocasión los señores de Ultimate nos han puesto las cosas mucho más difíciles, pues las diferentes piezas se encuentran separadas entre sí por largas distancias entre pasadizos llenos de peligros.

Sin embargo, no creáis que la cosa acaba aquí, que aún no hemos hecho apenas nada más que empezar. Ahora es cuando podremos empezar a recoger los cristales, algo que, como antes decíamos, puede ser cuestión de años. El luchar con todos esos meteoritos, rocas, artefactos mecánicos, esquivar las balas que repentinamente surgirán de cañones ocultos, y un sinfín de obstáculos más, va a resultar algo auténticamente demencial.

Pero si hemos conseguido llegar hasta aquí, lo que menos podemos hacer es desanimarnos: tan sólo nos quedan otros cuatro planetas más en los que tendre-









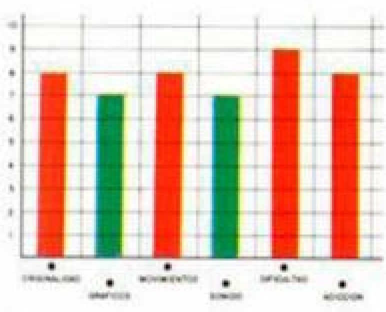
mos que repetir la operación casi por completo.

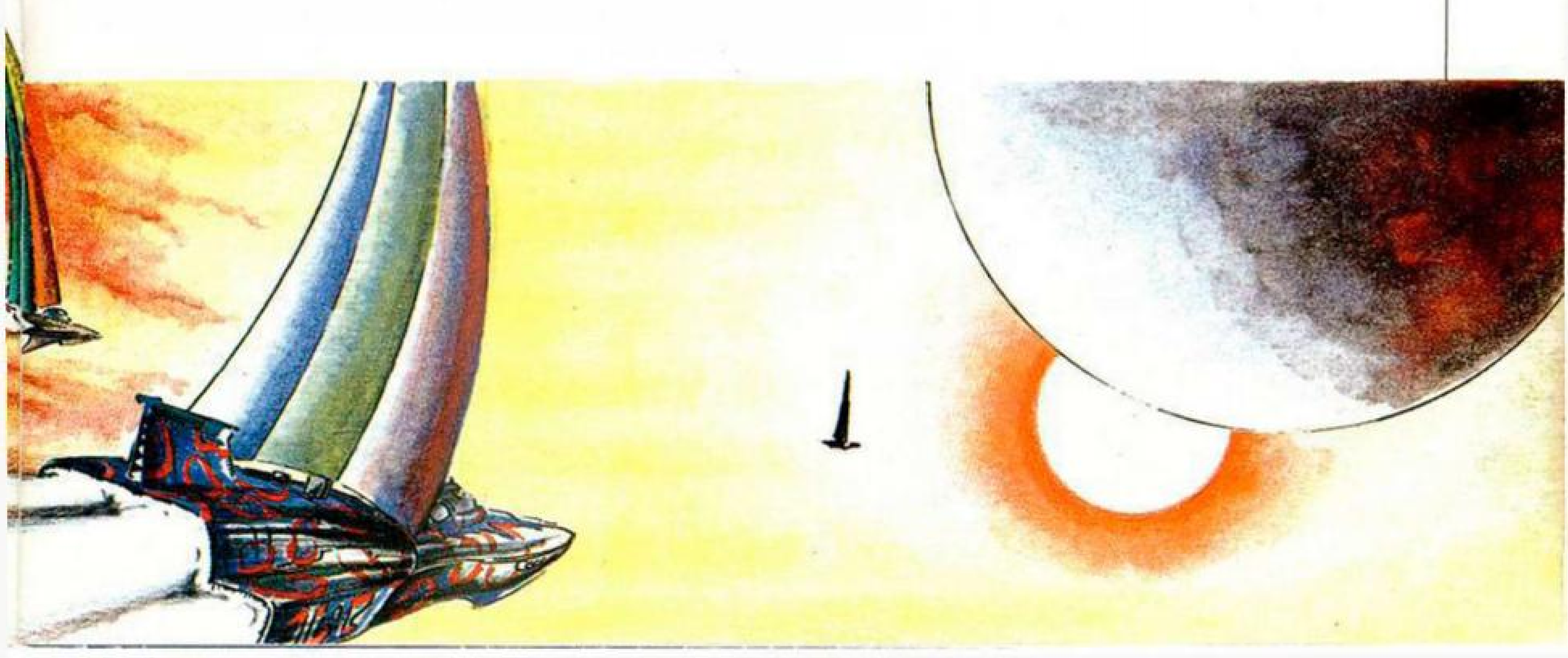
Cuando, seguramente gracias a un milagro, hayáis conseguido completar la aventura habiendo recogido todos los trozos de cybernita que están esparcidos por los cinco planetas, podréis decir con suma satisfacción, no sólo que sois una de las pocas personas que ha conseguido llegar hasta tan anhelado fin, sino también que tenéis más paciencia que el Santo Job y más moral que el Alcoyano.

Pero nada más lejos de nuestra intención que penséis que Cyberun es un programa malo con el que casi os vais a volver locos. Por supuesto que no. Al contrario, Cyberun es un excelente programa tanto por sus características gráficas como por su alto grado de adicción.

Por ejemplo en lo que a los gráficos se refiere, podemos decir que lo que más llama la atención es el movimiento tanto de la nave como de los enemigos, el cual, junto con el buen scroll de la pantalla, se puede decir que son sus detalles más relevantes, pues en cuanto al diseño en sí de los escenarios y los gráficos en general, quizá se echa un poco en falta algo más de variedad; pero pensando en la gran cantidad de pantallas que posee, tampoco es muy de extrañar el que no quede demasiada memoria para florituras.

En definitiva, Cyberun es un excelente programa, divertido y agradable cuyo único inconveniente reside en la gran dificultad que entraña el poder llegar a completar la aventura.





WINTER GAMES • Deportivo • Epyx

¡¡VIVE LA NIEVE!!

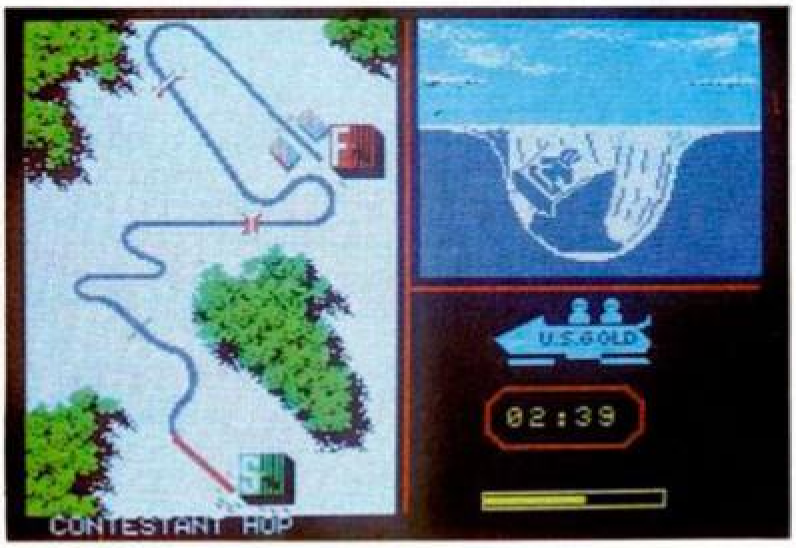
Nos encontramos en las blancas y frías pistas de Calgary, Alberta (Canadá), donde va a darse inicio dentro de breves instantes a la primera de las ocho pruebas que componen este I Campeonato Mundial de Deportes de Invierno para Ordenador. Los deportistas están ya en sus puestos, el público espera espectante, los jueces dan en estos momentos la señal de salida..., la competición ha comenzado.

No cabe duda que los programas deportivos son los que, si están bien realizados (por supuesto), despiertan un mayor interés en todos los usuarios.

Este tipo de juegos siempre ha encabezado los primeros puestos en todas las listas de éxitos, y con ellos han disfrutado miles y miles de personas: Match Point, Hypersports, The Way of the Exploding Fist, Bandera a cuadros, Match Day...

Esta relación se va haciendo cada vez más y más amplia y continúan apareciendo nuevos programas que nos ofrecen la posibilidad de disfrutar plenamente de toda la emoción que el deporte y la competición nos pueden ofrecer.







Así ha surgido este fantástico Winter Games, programa que ya está ocupando las primeras posiciones en los hits británicos y que ha sido creado y diseñado por Epyx, compañía de software cuyas creaciones estaban orientadas principalmente para el Commodore, pero que ahora se ha lanzado al mundo del Spectrum con juegos como Impossible Mission y este Winter Games.

Este juego nos traslada a tierras más frías, nos pone un gorrito, unos esquíes y ¡a retozar en la nieve! El programa completo está formado por siete pruebas diferentes relacionadas, por supuesto, con los deportes de invierno. Estas son: salto de esquí, esquí acrobático, velocidad sobre patines, patinaje artístico, bobsled, biathlon y patinaje libre estilo.

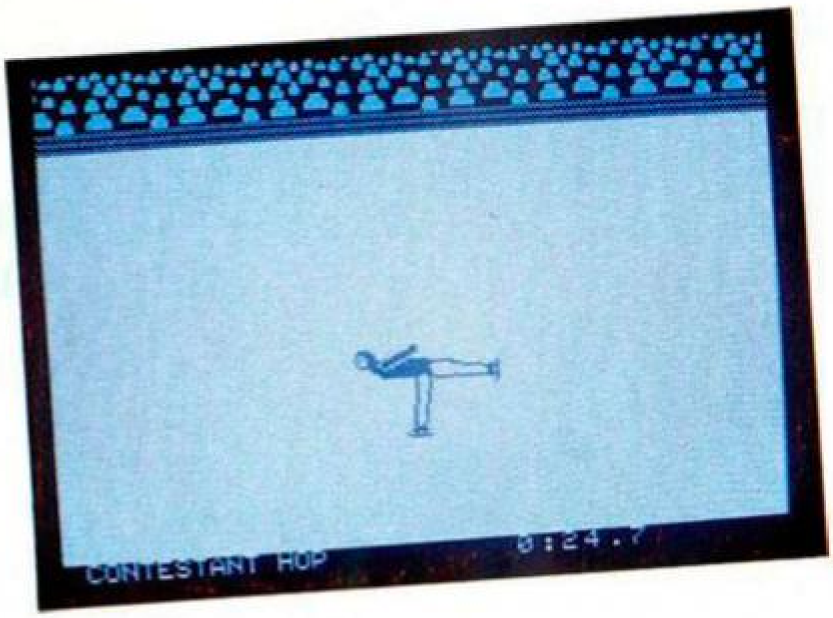
Salto de esquí: la primera prueba y una de las más emocionantes y divertidas. En ella deberemos saltar la mayor longitud posible, para lo cual deberemos intentar mantener en el aire la posición más aerodinámica que nos permita llegar más lejos. Aquí deberemos tener mucho cuidado pues el más mínimo fallo puede hacernos dar con nuestros huesos en la nieve.

Esquí acrobático: cantidad de divertido. Como en los anuncios de la tele, pero mucho más atractivo. Las piruetas que podemos dar son de lo más artístico si salen bien, pero tendremos que afinar mucho y no cometer ni el más ligero error si queremos conseguir una puntuación elevada.

Velocidad sobre patines: según dicen los patinadores de velocidad son las personas autopropulsadas que pueden alcanzar una mayor velocidad. Nosotros desde luego necesitaremos imprimir bastante rapidez al movimiento de nuestros dedos (o joystick en su defecto), para que nuestro corredor consiga ganar la carrera.

Patinaje artístico: en esta modalidad seguiremos necesitando de los patines, pero tendremos que cambiar de vestuario y ponernos una hermosa y ligera faldita que nos permita una







mayor facilidad de movimientos. Así podremos demostrar no sólo nuestra habilidad como patinadores, sino también nuestro sentido musical y del ritmo. Los movimientos que tendremos que realizar son obligatorios: camel spin, doble salto axel, triple salto lutz, etc...

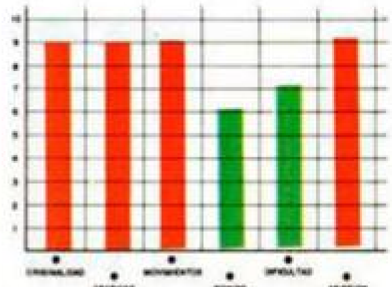
Bosbsled: quizá así, de primeras, no sepáis de qué va esto del bosbsled, pero seguro que habréis visto en algunas ocasiones a estos rápidos y aerodinámicos trineos que descienden a velocidades vertiginosas por unos estrechos caminos de hielo. Eso es el bosbsled. Preparaos a alcanzar velocidades de hasta 90 millas por hora.

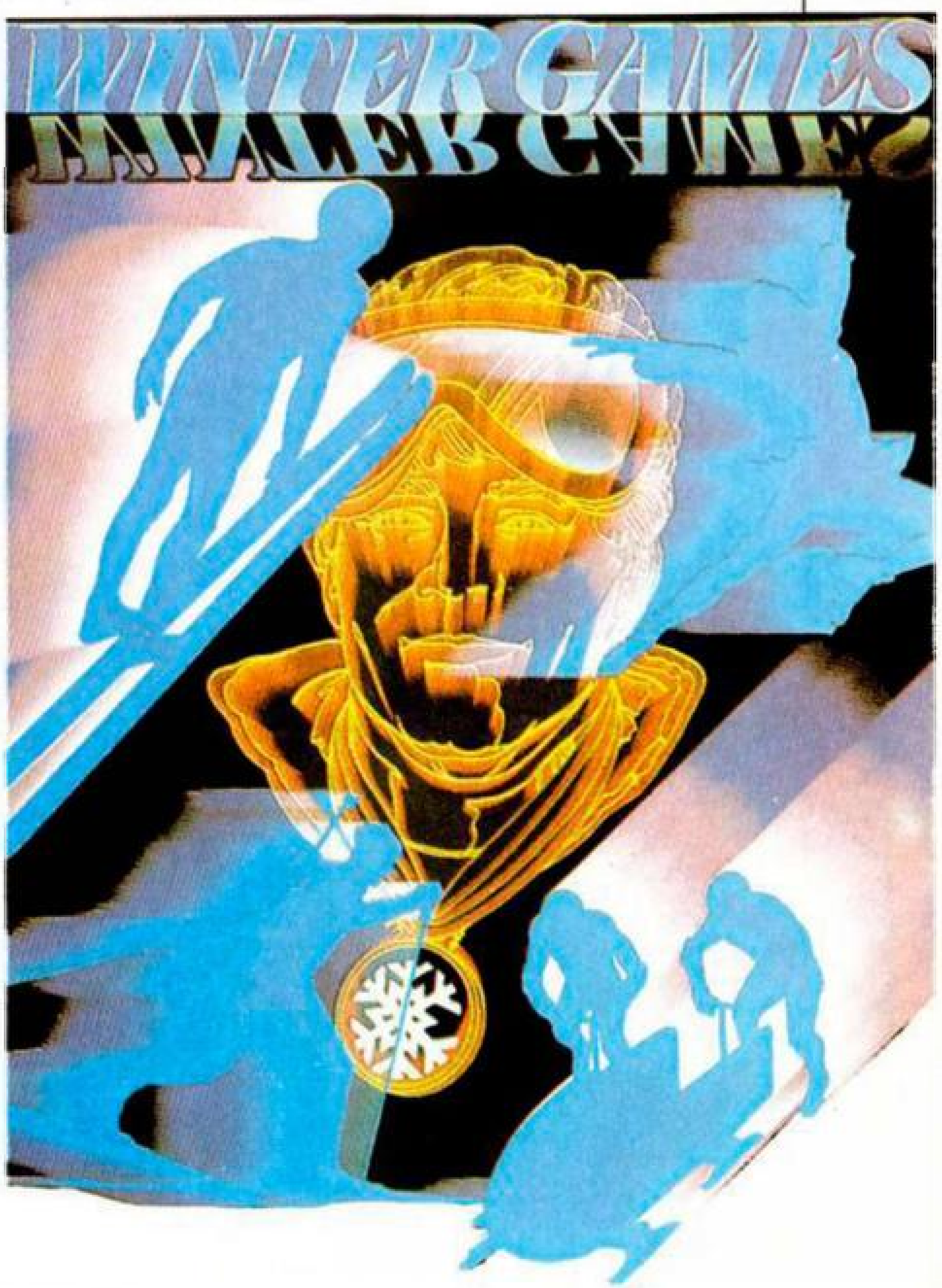
Biathlon: ésta es una de las pruebas más completas y variadas, ya que se compone de esquí de fondo, es decir, una especie de carrera de resistencia con esquíes, con la particularidad de que cada cierto tiempo deberemos detenernos a realizar unos cuantos tiros con carabina. Dependiendo del tiempo invertido en completar el recorrido y de las dianas obtenidas, así conseguiremos una mayor o menor puntuación.

Patinaje libre estilo: esta última prueba es muy similar a la de patinaje artístico, con la particularidad de que las piruetas y saltos que realicemos no deberán cumplir ninguna norma específica y tan sólo contará nuestra vocación artística, gracia, salero y estilo.

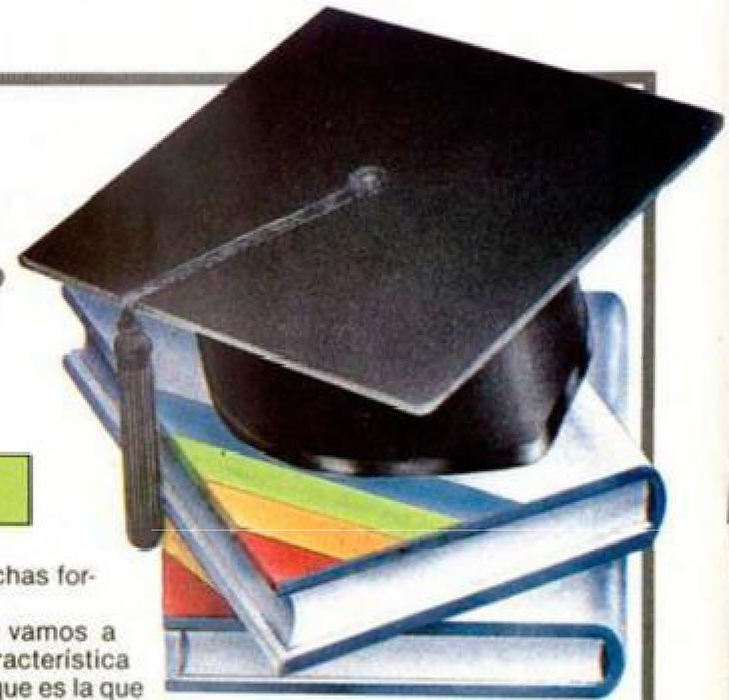
Como podréis comprobar, mayor variedad no se le puede pedir a un programa y el interés que despierta cada una de las pruebas es más que notable. Es una verdadera delicia a todos los niveles.

Posiblemente éste sea uno de los juegos más atractivos de los vistos en las últimas fechas, no sólo por la diversión que nos ofrecen todas y cada una de las pruebas, sino también por el gran realismo con el que están realizadas, por lo genial de sus gráficos y presentación de pantallas, por lo auténtico de los movimientos de los deportistas y porque..., ¡vamos, que es todo un juegazo!





Profesor particular



Arturo LOBO y J. J. LEON

PLANO AFIN (1)

El programa de esta semana está pensado para ayudaros con los problemas de rectas en el plano afín, una parte del temario de BUP. En próximas semanas os presentaremos otro, esta vez sobre cambios de base y diversos problemas más. De momento, vamos a explicaros qué hace éste y cómo manejarlo.

El programa proporciona las ecuaciones de una recta en cinco formas: paramétrica, implícita, segmentaria y continua, una vez se define la recta de que se trate por uno de estos cuatro métodos: dando dos puntos, un punto y un vector de dirección, un punto y la pendiente o los segmentos que define sobre los ejes. Otra aplicación práctica sería el cambio de ecuación. Por ejemplo, dada la forma explícita bien podemos hallar un punto y la pendiente, con lo que, entrando en el menú, podremos obtener cualquiera de las otras formas de ecuaciones de la recta. El programa es simple de manejar y probablemente le sacaréis partido de muchas formas.

Por último, vamos a explicar su característica fundamental, que es la que ha hecho que el listado se alargara un poco. Consiste en que, como en alguna otra ocasión, el programa puede manejar números enteros y fracciones indistintamente, y no sólo en los resultados, sino también en los INPUTS. Para introducir una fracción debéis teclear numerador, raya de fracción y denominador, y nunca decimales (por ejemplo 17/5, —3/2). Hemos buscado con ello, como siempre, que el programa opere de la misma forma en que operáis vosotros, para que con él podáis aprender rápidamente lo que en clase no siempre resulta tan fácil.

```
   2 CLEAR 64999: GO SUB 8000: RESTORE : DIM a(2): BORDER 6: PAPER 6
  80 LET frac=7000: DIM d(3,2): DIM p(2,2,2): DIM e(2,2): DIM i(3,2): DIM s(2,2): DIM c(2,2,2)
  85 GO TO 1600
  90 DEF FN a()=CODE SCREEN$ (24-PEEK 23689,32-PEEK 23688)
 100 CLS : RESTORE 100: PRINT AT 0,4;"ECUACIONES DE LA RECTA"
 110 FOR i=1 TO 5: READ linea,d$: PRINT AT linea,0;d$: NEXT i
 120 DATA 3,"Parametrica",7,"Explicita",10,"Implicita",14,"Segmentaria",19,"Continua"
 125 PLOT 100,135: DRAW 0,24: FOR i=1 TO 2: READ p$,ind
 130 PRINT AT 2*ind,13;p$;: LET a(1)=p(ind,1,1): LET a(2)=p(ind,1,2): IF a(1) THEN GO SUB frac: GO SUB 9990: PRINT "t";
 140 LET a(1)=p(ind,2,1): LET a(2)=p(ind,2,2): IF a(1)/a(2)>0 THEN PRINT "+";
 150 IF a(1) THEN GO SUB frac
 160 NEXT i
 170 DATA "x=",1,"y=",2
 180 PRINT AT 7,12;"y=";: LET a(1)=e(1,1): LET a(2)=e(1,2): IF a(1) THEN GO SUB frac: GO SUB 9990: PRINT "x";
 190 LET a(1)=e(2,1): LET a(2)=e(2,2): IF a(1)/a(2)>0 THEN PRINT "+";
 200 IF a(1) THEN GO SUB frac
 205 PRINT AT 10,12;
 210 LET a(1)=i(1,1): LET a(2)=i(1,2): IF a(1) THEN GO SUB frac: GO SUB 9990: PRINT "x";
 220 IF NOT i(2,1) THEN GO TO 240
 230 PRINT "+";: LET a(1)=i(2,1): LET a(2)=i(2,2): PRINT (CHR$ 8 AND SGN (a(1)*a(2))<0);: GO SUB frac: GO SUB 9990: PRINT "y";
 240 PRINT ("+" AND i(3,1)/i(3,2)>0);: LET a(1)=i(3,1): LET a(2)=i(3,2): IF a(1)<>0 THEN GO SUB frac
 242 PRINT "=0"
 245 LET prin=0: LET lug=13
 250 FOR i=1 TO 2: LET a(1)=s(i,1): LET a(2)=s(i,2): PRINT AT 15,lug;: GO SUB frac
 260 LET lon=(6*len+10)*(a(2)<>1)+(8+8*LEN STR$ a(1))*(a(2)=1)+8*(s(i,1)*s(i,2)<0): PLOT 100+prin*8,50: DRAW lon,0: PRINT AT 13,lug+INT (lon/16)-1;CHR$ (119+i);
 270 LET prin=2+INT ((lon+6)/8): LET lug=lug+prin: PRINT AT 14,lug-2;CHR$ (25+i*18);
 275 NEXT i: PRINT 1
 280 FOR i=1 TO 2: FOR j=1 TO 2
 290 LET c1=LEN STR$ ABS c(i,j,1): LET c2=LEN STR$ ABS c(i,j,2)
 300 LET d(i,j)=c1+(c2-c1)*(c2>c1)*(i=2 AND c(i,j,1)/c(i,j,2)<0): IF c(i,j,2)<>1 THEN LET d(i,j)=INT ((i-1+6*d(i,j))/8)+1
 310 LET d(i,j)=d(i,j)+2*(i=1)+(i=2)*(c(i,j,1)*c(i,j,2)<0)
 320 NEXT j: NEXT i
 330 FOR i=1 TO 2: LET d(3,i)=d(1,i)+(d(2,i)-d(1,i))*(d(2,i)>d(1,i)): NEXT i
 340 PLOT 100,19: DRAW (1+d(3,1))*8,0: DRAW INVERSE 1;16,0: DRAW (1+d(3,2))*8,0: PRINT AT 19,14+d(3,1);"="
 350 FOR i=1 TO 2: FOR j=1 TO 2: FOR k=1 TO 2: LET a(k)=c(i,j,k)
 360 NEXT k: LET P(i)=13+(3+d(3,j))*(j=2)+(d(3,j)-d(i,j))/2
 370 IF i=1 THEN PRINT AT 18,p(i);CHR$ (119+j);"-" AND a(1)/a(2)>0;: GO SUB frac
 380 IF i=2 THEN PRINT AT 20,p(i);: GO SUB frac
 390 NEXT j: NEXT i
 395 GO SUB 500: GO TO 1600
 399 STOP
 410 LET a1=(ABS a(1) AND ABS a(1)>ABS a(2))+(ABS a(2) AND ABS a(2)>ABS a(1))
 420 LET a2=ABS a(1)+ABS a(2)-a1
 425 IF NOT a2 THEN RETURN
 430 LET rr=a1-a2*INT (a1/a2)
 440 IF rr=0 THEN LET a(1)=a(1)/a2: LET a(2)=a(2)/a2: RETURN
 450 LET a1=a2: LET a2=rr: GO TO 430
 500 PRINT #0;AT 1,8;"pulsa una tecla": PAUSE 0: PRINT #0;AT 1,8;TAB 24: RETURN
 600 FOR k=1 TO LEN d$: IF d$(k)="/" THEN LET a(1)=VAL d$( TO k-1): LET a(2)=VAL d$(k+1 TO ): GO SUB 400: GO TO 620
 610 NEXT k: LET a(1)=VAL d$: LET a(2)=1
 620 RETURN
1100 FOR i=1 TO 2: FOR j=1 TO 2: FOR k=1 TO 2
1110 LET c(i,j,k)=p(j,3-i,k)*(-1 OR (i<>2 AND k=1))
1120 NEXT k: NEXT j: NEXT i
1130 IF flag<>3 THEN GO TO 1500
1140 RETURN
1200 LET s(2,1)=e(2,1): LET s(2,2)=e(2,2)
1210 LET s(1,1)=-e(2,1)*e(1,2)
1220 LET s(1,2)=e(1,1)*e(2,2)
1230 IF flag<>1 THEN GO TO 1400
1240 RETURN
1300 LET e(1,1)=-i(1,1)*i(2,2)
1310 LET e(1,2)=i(1,2)*i(2,1)
1320 LET e(2,1)=-i(3,1)*i(2,2)
1330 LET e(2,2)=i(3,2)*i(2,1)
1340 IF flag<>4 THEN GO TO 1200
1350 RETURN
1400 FOR i=1 TO 2: LET p(1,2,i)=0: LET p(1,1,i)=-s(1,i)
1410 LET p(2,1,i)=s(2,i): LET p(2,2,i)=s(2,i)
1420 NEXT i: LET p(1,1,2)=s(1,2): LET p(1,2,2)=1
1430 IF flag<>5 THEN GO TO 1100
1440 RETURN
1500 FOR i=1 TO 2: LET i(1,i)=c(2,2,i): LET i(2,i)=c(2,1,i)*(-1 OR i=2): NEXT i
1510 LET a(2)=1: FOR i=1 TO 2: FOR j=1 TO 2: LET a(2)=a(2)*c(i,j,2): NEXT j: NEXT i
1520 LET a(1)=c(1,1,1)*c(2,2,1)*c(1,2,2)*c(2,1,2)-c(1,2,1)*c(2,1,1)*c(1,1,2)*c(2,2,2)
1530 GO SUB 400: LET i(3,1)=a(1): LET i(3,2)=a(2)
1540 IF flag<>2 THEN GO TO 1300
1550 RETURN
1600 CLS : PRINT AT 3,12; INVERSE 1;"MENU": RESTORE 1620: FOR i=1 TO 4: READ a$: PRINT AT 4+2*i,6;i;" ";a$: NEXT i
1620 DATA "Punto y vector","Dos puntos","Punto y pendiente","Segmentos en los ejes"
1630 GO SUB 500: IF CODE INKEY$<49 OR CODE INKEY$>52 THEN GO TO 1630
1640 GO SUB 1600+100*VAL INKEY$
1650 GO SUB 1000+100*flag
1660 GO TO 100
1700 FOR i=2 TO 1 STEP -1
1710 GO SUB 2200-100*i: LET p(1,1,1)=a(1): LET p(1,1,2)=a(2): LET d$=e$: GO SUB 600: LET p(2,i,1)=a(1): LET p(2,i,2)=a(2)
1720 NEXT i: LET flag=1: RETURN
1800 FOR i=1 TO 2
1810 GO SUB 2100: LET p(i,1,1)=a(1): LET p(i,1,2)=a(2): LET d$=e$: GO SUB 600: LET p(2,i,1)=a(1): LET p(2,i,2)=a(2)
1820 NEXT i: FOR i=1 TO 2
1830 LET p(i,1,1)=p(i,1,1)*p(i,2,2)-p(i,2,1)*p(i,1,2): LET p(i,1,2)=p(i,1,2)*p(i,2,2)
1840 NEXT i: LET flag=1: RETURN
1900 INPUT "Pendiente: ";LINE d$: GO SUB 600: LET e(1,1)=a(1): LET e(1,2)=a(2): GO SUB 2100
1910 LET e(2,1)=a(1)*e(1,1): LET e(2,2)=a(2)*e(1,2): LET d$=e$: GO SUB 600: LET e(2,1)=a(1)*e(2,2)-a(2)*e(2,1): LET e(2,2)=a(2)*e(2,2): LET flag=2: RETURN
2000 FOR i=1 TO 2: INPUT "segmento sobre el eje ";(CHR$ (119+i));"? ";LINE d$: GO SUB 600: LET s(i,1)=a(1): LET s(i,2)=a(2)
2010 NEXT i: LET flag=4: RETURN
2100 INPUT "Coordenadas del punto";"x= ";LINE d$;"y= ";LINE e$
2110 GO SUB 600: RETURN
2200 INPUT "Componentes del vector";"x= ";LINE d$,"y= ";LINE e$
2210 GO SUB 600: RETURN
7000 GO SUB 400: IF ABS a(2)=1 THEN PRINT a(1)*SGN a(2);: LET len=8*LEN STR$ (a(1)*SGN a(2)): GO TO 7130
7005 PRINT ("-" AND a(1)*a(2)<0);: LET a(1)=ABS a(1): LET a(2)=ABS a(2)
7010 LET a1=LEN STR$ a(1): LET a2=LEN STR$ a(2)
7020 LET len=a1+(a2-a1)*(a2>a1)
7030 DIM c$(len): DIM b$(len)
7040 LET c$(len-a1+1 TO len)=STR$ a(1): LET b$(len-a2+1 TO len)=STR$ a(2)
7050 LET p2=-21+8*PEEK 23689
7060 FOR l=0 TO len-1
7070 LET p1=(33-PEEK 23688)*8+1+6*l
7080 PLOT p1,p2: DRAW 6,0
7090 IF a2>=len-l THEN LET a3=VAL b$(l+1): RANDOMIZE FN p(p1+1-3*(len-a2),p2-7,a3)
7100 IF a1>=len-l THEN LET a4=VAL c$(l+1): RANDOMIZE FN p(p1+1-3*(len-a1),p2+1,a4)
7110 NEXT l
7120 FOR l=1 TO INT ((-1+6*len)/8)+1: PRINT OVER 1;" ";: NEXT l
7130 RETURN
8000 RESTORE 8050
8010 FOR i=65000 TO 65077: READ a: POKE i,a: NEXT i
8020 FOR i=65400 TO 65431: READ a: POKE i,a: NEXT i
8030 FOR i=65480 TO 65527 STEP 4: FOR j=0 TO 2: READ a: POKE i+j,a: NEXT j: POKE i+3,0: NEXT i
8050 DATA 205,148,30,245,205,153,30,197,205,153,30,197,205,200,183,198,5,95,241,229,38,255,203
8060 DATA 39,203,39,198,200,111,78,44,86,44,70,225,96,121,205,120,255,205,120,255,121,205,120
8070 DATA 255,205,120,255,124,205,120,255,120,198,5,6,0,79,205,43,45,1,1,0,205,43,45,1,1,0,197,205,43,45,193,201
8080 DATA 6,4,197,203,127,40,11,87,77,245,213,229,205,229,34,225,209,241,44,203,7,193,16,234,79,108,214,4,111,181,201
8090 DATA 105,153,96,30,34,32,105,30,240,105,41,96,36,111,32,246,225,224,104,233,96,241,38,64,105,105,96,105,113,96,0,0,32,0,112,0
8100 DEF FN p(x,y,n)=x+y+n+USR 65000
8110 RETURN
9990 IF FN a()=49 THEN PRINT CHR$ 8;
9992 RETURN
```

SOLUCION A LOS EJERCICIOS

SOLUCIONES A LOS EJERCICIOS

1.- Habrá que cambiar las siguientes líneas:

```
70        LD   HL,20479   ; Final del segundo tercio.
80        LD   C,64       ; 64 scans en un tercio.
240       LD   HL,18432   ; Inicio del segundo tercio.
250       LD   C,64       ; 64 scans en un tercio.
```

2.- Habrá que modificar la rutina, a partir de "FIN_1" de la siguiente forma:

```
810 FIN_1 PUSH DE         ; Preserva "DE".
820       LD   A,(23681)  ; Carga variable.
830       BIT  2,A        ; Comprueba el flag.
840       JR   NZ,FIN_2   ; Si es "1", salta a "FIN_2".
850       LD   HL,23296   ; Borra el scan que se había
860       LD   DE,23297   ; almacenado en el buffer de
870       LD   BC,31      ; impresora.
880       XOR  A
890       LD   (HL),A
900       LDIR
910 FIN_2 POP  DE         ; Recupera "DE"
920       LD   HL,23296   ; Sigue como antes de la
930       LD   BC,32      ; modificación.
940       LDIR
950       RET
```

3.- El punto de entrada por "SCRA_1" corresponde al scroll ascendente, y por "SCRA_2" al scroll descendente.

```
100 SCRA_1 LD  HL,22528   ; Transfiere los primeros
110        LD  DE,23296   ; 32 bytes del archivo de atri-
120        LD  BC,32      ; butos a los 32 primeros del
130        LDIR           ; buffer de impresora.
140        LD  DE,22528   ; Transfiere, 32 bytes hacia
150        LD  BC,768     ; atras, el archivo de atributos
160        LDIR           ; más los 32 primeros bytes del
170        RET            ; buffer de impresora y retorna.
180 SCRA_2 LD  HL,23295   ; Transfiere, 32 bytes hacia
190        LD  DE,23327   ; delante, el archivo de atribu-
200        LD  BC,768     ; tos entrando en los 32 primeros
210        LDDR           ; bytes del buffer de impresora.
220        LD  HL,23327   ; Transfiere los 32 primeros
230        LD  BC,32      ; bytes del buffer de impresora
240        LDDR           ; a los 32 primeros de atributos.
250        RET            ; Retorna.
```

Aprovechamos la circunstancia de que el buffer de impresora se encuentra inmediatamente a continuación del archivo de atributos.

dos huecos que nos quedan libres.

280: Finalmente, los mezclamos con lo que teníamos en «B».

A la salida de la rutina, tendremos en «A» el número de scan, comprendido entre «0» y «191». El proceso se entiende perfectamente si se va mirando la Figura 10-7 al mismo tiempo.

Vamos ahora con la segunda rutina, en este caso, tendremos que sintetizar, en «HL», la dirección correspondiente a un scan dado. Escribiremos la rutina de forma que no se alteren los cinco bits inferiores de «L» ya que son los que definen el número de columna. Suponemos que el número de scan se encuentra en el registro «A»; aunque de nuevo, utilizaremos el «B» como almacenamiento temporal:

```
380 SIG_2 LD   B,A
390       LD   H,#40
400       AND  #07
410       OR   H
420       LD   H,A
430       LD   A,B
440       AND  #C0
450       SRL  A
460       SRL  A
470       SRL  A
480       OR   H
490       LD   H,A
500       LD   A,L
510       AND  #1F
520       LD   L,A
530       LD   A,B
540       AND  #38
550       SLA  A
560       SLA  A
570       OR   L
580       LD   L,A
```

De la misma forma que antes, vamos a explicar, línea a línea, el funcionamiento de esta rutina:

380: Guardamos en «B» el contenido de «A» para que no sea destruido al operar.

390: Cargamos 40h en «H» para poner el bit 6 a «1» y los restantes a «0» (recuerde que 40h = = 01000000b).

400: Aislamos los tres bits de menos peso de «A».

410: Los mezclamos con el contenido de «H».

420: Y almacenamos el resultado en «H».

430: Ahora, volvemos a recuperar el número de scan desde «B».

440: Esta vez, aislamos los dos bits de más peso con la máscara C0h (obsérvese el frecuente uso que hacemos de las máscaras en estas rutinas).

450: En esta línea y las dos siguientes, rotamos estos dos bits, tres lugares a la derecha.

480: Los mezclamos con el contenido de «H».

490: Y almacenamos el resultado en «H». Ya tenemos completo el byte alto de nuestra dirección.

500: Cargamos en «A» el byte bajo.

510: Borramos los tres bits superiores, dejando los cinco inferiores inalterados.

520: Reponemos el contenido de «L». Ahora, «L» sólo contiene los cinco bits que definen la columna.

530: Recuperamos el número de scan desde «B».

540: Ahora, aislamos los bits 3, 4 y 5.

550: En esta línea y la si-

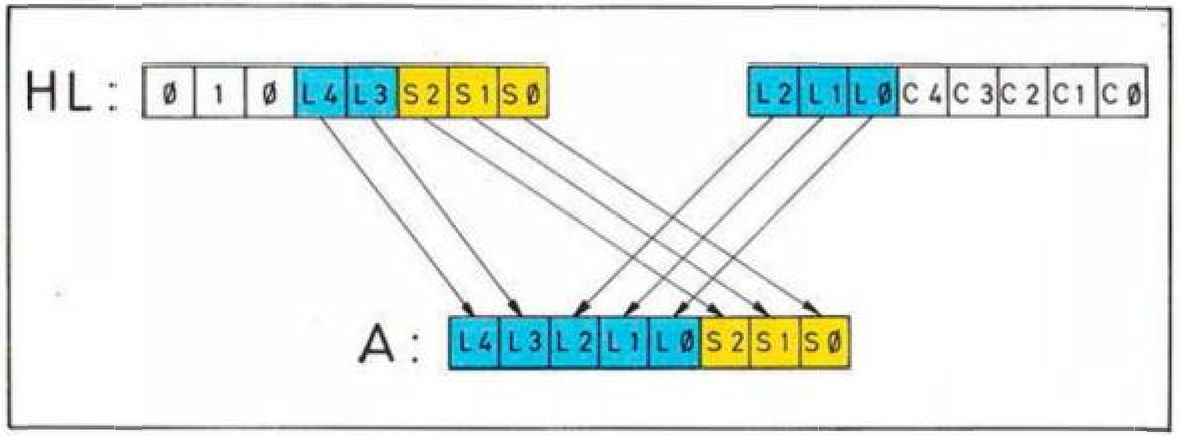


Fig. 10-7. Síntesis del número de scan a partir de la dirección de la pantalla.

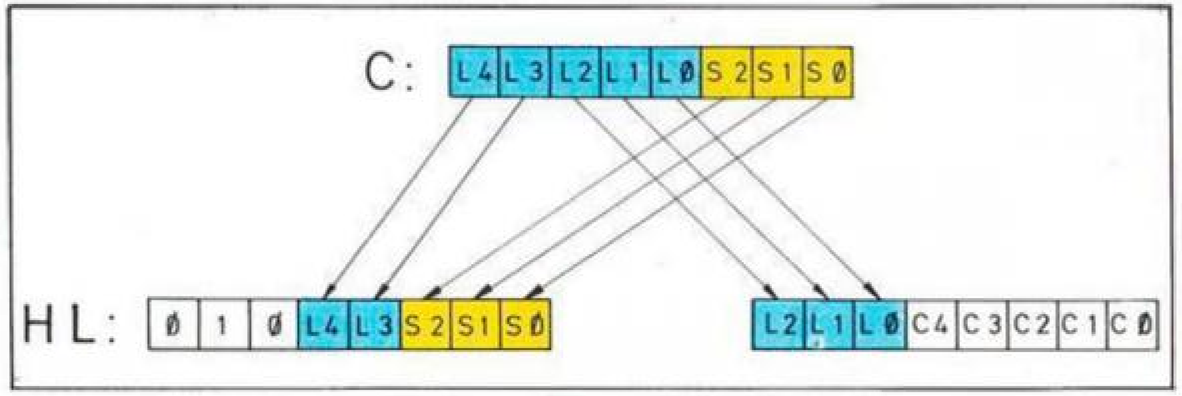


Fig. 10.8. Síntesis de la dirección de pantalla a partir del número de scan.

guiente, los rotamos dos lugares a la izquierda.

570: Los mezclamos con el contenido de «L».

580: Y finalmente, almacenamos el resultado en «L» con lo que queda completo el byte bajo de la dirección.

De nuevo, el procedimiento se entiende mejor consultando la Figura 10-8. A la salida de la rutina, tendremos en «HL» una dirección de pantalla correspondiente al scan cuyo número estaba en «A» y a la misma columna que definían los cinco bits inferiores de «L». Como detalle adicional, todavía tendremos en «B» el número de scan por si pudiera ser útil.

Ya tenemos las dos rutinas que necesitábamos, la primera que llama «SIG-1» y la segunda «SIG-2». Volvamos al problema inicial; se trataba de hallar la dirección del scan siguiente o del anterior a una dirección dada. Para hallar el scan siguiente, primero utilizamos «SIG-1», luego incrementamos el número de scan y finalmente, utilizamos «SIG-2». En el caso de tener que hallar la dirección del scan anterior, utilizamos el mismo procedimiento salvo que decrementamos el número de scan en lugar de incrementarlo. En ambos casos, deberemos escribir las rutinas de forma que salgan con el indicador de acarreo a «1» si la dirección es correcta, y a «Ø» si cae fuera de la pantalla.

En los dos casos, tendremos que utilizar las rutinas «SIG-1» y «SIG-2» por lo que no parece muy adecuado escribir dos rutinas, una para hallar el scan siguiente y otra para el anterior. Lo mejor es escribir una sola rutina que nos dé la dirección del scan siguiente o del anterior en función del contenido de un flag. Utilizaremos el bit Ø del registro «C» como flag, la rutina tendrá tres puntos de entrada: Si entramos por «SIG», nos dará la dirección del octeto del scan siguiente a aquél cuya dirección contenga «HL». Si entramos por «ANT», nos dará la dirección del octeto del scan anterior. Finalmente, podremos entrar por «SIG-1» habiendo fijado, previamente, el bit Ø de «C» a «1» si queremos la dirección siguiente, o a «Ø» si queremos la anterior. Estas serían las líneas que habría que añadir a «SIG-1» y «SIG-2» para tener la rutina completa:

```
11Ø SIG      SET  Ø,C
12Ø          JR   SIG_1
13Ø ANT      RES  Ø,C
========================
14Ø SIG_1    ..........
...          ..........
========================
29Ø          BIT  Ø,C
3ØØ          JR   Z,ANT_1
31Ø          INC  A
32Ø          CP   #CØ
33Ø          PUSH AF
34Ø          JR   SIG_2
35Ø ANT_1    SUB  1
36Ø          CCF
37Ø          PUSH AF
========================
38Ø SIG_2    ..........
...          ..........
========================
59Ø          POP  AF
6ØØ          RET
```

Vamos a explicar la rutina línea a línea:

11Ø: Ponemos a «1» el flag para que la dirección que hallemos sea la del scan siguiente.

12Ø: Saltamos a «SIG-1» (línea 14Ø).

13Ø: Ponemos a «Ø» el flag para que la dirección


```
55Ø86 25Ø SIG_1  LD    A,H
55Ø87 26Ø        AND   #Ø7
55Ø89 27Ø        LD    B,A
55Ø9Ø 28Ø        LD    A,H
55Ø91 29Ø        AND   #18
55Ø93 3ØØ        SLA   A
55Ø95 31Ø        SLA   A
55Ø97 32Ø        SLA   A
55Ø99 33Ø        OR    B
551ØØ 34Ø        LD    B,A
551Ø1 35Ø        LD    A,L
551Ø2 36Ø        AND   #EØ
551Ø4 37Ø        SRL   A
551Ø6 38Ø        SRL   A
551Ø8 39Ø        OR    B
551Ø9 4ØØ        BIT   Ø,C
55111 41Ø        JR    Z,ANT_1
55113 42Ø        INC   A
55114 43Ø        CP    #CØ
55116 44Ø        PUSH  AF
55117 45Ø        JR    SIG_2
55119 46Ø ANT_1  SUB   1
55121 47Ø        CCF
55122 48Ø        PUSH  AF
55123 49Ø SIG_2  LD    B,A
55124 5ØØ        LD    H,#4Ø
55126 51Ø        AND   #Ø7
55128 52Ø        OR    H
55129 53Ø        LD    H,A
5513Ø 54Ø        LD    A,B
55131 55Ø        AND   #CØ
55133 56Ø        SRL   A
55135 57Ø        SRL   A
55137 58Ø        SRL   A
55139 59Ø        OR    H
5514Ø 6ØØ        LD    H,A
55141 61Ø        LD    A,L
55142 62Ø        AND   #1F
55144 63Ø        LD    L,A
55145 64Ø        LD    A,B
55146 65Ø        AND   #38
55148 66Ø        SLA   A
5515Ø 67Ø        SLA   A
55152 68Ø        OR    L
55153 69Ø        LD    L,A
55154 7ØØ        POP   AF
55155 71Ø        POP   BC
55156 72Ø        POP   DE
55157 73Ø        JR    NC,FIN_1
55159 74Ø        PUSH  HL
5516Ø 75Ø        PUSH  BC
55161 76Ø        LD    BC,32
55164 77Ø        LDIR
55166 78Ø        POP   BC
55167 79Ø        POP   HL
55168 8ØØ        JR    BUC_3
5517Ø 81Ø FIN_1  LD    HL,23296
55173 82Ø        LD    BC,32
55176 83Ø        LDIR
55178 84Ø        RET

Pass 2 errors: ØØ
```

Fig. 10-10. Listado completo de la rutina scroll vertical.

EJERCICIOS

1.- Modifique las rutinas de scroll a derecha e izquierda para que trabajen, exclusivamente, sobre el segundo tercio de pantalla, es decir, el central.

2.- Modifique la rutina de scroll vertical de forma que lo que se pierda por un extremo de la pantalla (arriba o abajo), no aparezca por el contrario (abajo o arriba) si el bit 2 de la dirección 23681 es "Ø" (scroll lineal) y funcione normalmente si este bit es "1" (scroll esférico).

3.- Escriba dos rutinas que realicen el scroll vertical en el archivo de atributos. Una de las rutinas deberá hacer el scroll hacia arriba y la otra, hacia abajo. El scroll ha de ser "esférico".

das del bucle vamos tratando los scans uno a uno hasta que terminemos con toda la pantalla, momento en el que salimos por «FIN_1».

En la línea 810 («FIN_1»), cargamos en «HL» la dirección inicial del buffer de impresora que será el origen de la última transferencia. El destino será el último scan que hayamos procesado, cuya dirección se encuentra ya en «DE» por lo que no es necesario cargarla. En 820 cargamos, de nuevo, la longitud en «BC»; no preservamos los registros porque es la última transferencia y ya no nos importa que se pierdan sus contenidos. Finalmente, realizamos la transferencia en la línea 830 y retornamos en 840.

Esta vez, se pone más complicado el modificar la rutina para que el scroll no sea «esférico»; el procedimiento consiste en borrar el buffer de impresora antes de efectuar la última transferencia. La modificación no es demasiado difícil de realizar. No hemos querido privar al lector del placer de realizarla por sí mismo, así que este será otro de los ejercicios del capítulo. De nuevo, le recomendamos que intente hacerlo antes de mirar la solución.

Sólo queda ensamblar la rutina y probarla. Si va a hacerlo a mano, le recomendamos que la ensamble en hexadecimal y compruebe su resultado comparando con las líneas 370 a 480 del Programa-1 donde se muestra un ejemplo de cómo utilizar estas rutinas para realizar un scroll total de la pantalla.

Este programa es similar al que ilustraba el prólogo; lo único que hemos hecho es añadir la rutina de scroll vertical y modificar algo el Basic para poder manejarla. Le recomendamos que cargue y ejecute este programa ya que será una buena forma de familiarizarse con el uso de estas rutinas. Si va a usarlas en un programa suyo, recuerde que el contenido de «BC» en el retorno será «0», por lo que llamarlas con «RANDOMIZE USR ...» equivale a hacer un «RANDOMIZE 0», es decir, copiar el contador de «FRAMES» en la variable «SEED», lo cual no siempre resulta conveniente; en ese caso, tal vez sea mejor que las llame con «LEZ z=USR ...».

También es importante señalar que estas rutinas hacen el scroll solamente del fichero de pantalla, no del de atributos. Este último es bastante más fácil de «scrolar» ya que las direcciones correspondientes a líneas consecutivas, son también consecutivas. De nuevo, es un placer que tampoco queremos arrebatar al lector, así que lo dejamos para los ejercicios.

Hasta aquí, hemos visto las instrucciones de manejo de bits. En el próximo capítulo, veremos las que se utilizan para llamar a subrutinas, aprenderemos algo más de la pila y, en los ejemplos, terminaremos el «procesador de pantallas» que, estamos seguros, resultará muy útil para que el lector lo incluya en sus propios programas. Ahora, le recomendamos que intente resolver los ejercicios antes de pasar al capítulo siguiente.

```
*HISOFT GENS3M ASSEMBLER*
      ZX SPECTRUM

  Copyright HISOFT 1983
  CURSO C/M MICROHOBBY

Pass 1 errors: 00

        50 *C-
        60 *D+
        70 ;
        80 ;SCROLL_VERTICAL
        90 ;
55060 100           ORG     55060
55060 110 SCR_3     SET     0,C
55062 120           LD      HL,16384
55065 130           JR      SCR
55067 140 SCR_4     RES     0,C
55069 150           LD      HL,22496
55072 160 SCR       PUSH    HL
55073 170           LD      DE,23296
55076 180           PUSH    BC
55077 190           LD      BC,32
55080 200           LDIR
55082 210           POP     BC
55083 220           POP     HL
55084 230 BUC_3     PUSH    HL
55085 240           PUSH    BC
```


sea, esta vez, la correspondiente al scan anterior.

140 a 280: RUTINA «SIG-1».
290: Comprobamos el flag.
300: Si es «0», saltamos a «ANT-1» (línea 350).
310: Incrementamos el número de scan.
320: Lo comparamos con 192 para poner a «1» el acarreo sólo si es menor.
330: Guardamos el estado actual de los indicadores (el único que nos interesa es el de acarreo).
340: Saltamos a «SIG-2» (línea 380).
350: Decrementamos el número de scan. Utilizamos «SUB 1» en lugar de «DEC a» ya que esta última no afecta al indicador de acarreo.
360: Complementamos el indicador de acarreo para que sólo valga «1» si la dirección es correcta.
370: Guardamos el estado de los indicadores.
380 a 580: RUTINA «SIG-2».
590: Recuperamos los indicadores para recuperar el de acarreo.
600: Retornamos.

A la salida de esta rutina, tendremos en «HL» la nueva dirección que podremos considerar correcta si el indicador de acarreo se encuentra a «1»; en caso contrario, sabremos que nos hemos salido de la pantalla. También, tendremos en «A» el número de scan que, de momento, no vamos a utilizar pero que alguna vez puede resultarnos útil.

De nuevo, le recomendamos que ensamble la rutina en hexadecimal y utilice el formato del «CARGADOR UNIVERSAL». Deberá quedarle algo así:

```
1 CBC11802CB817CE60747 1186
2 7CE618CB27CB27CB27B0 1280
3 477DE6E0CB3FCB3FB0CB 1561
4 4128063CFEC0F51804D6 1104
5 013FF5472640E607B467 1002
6 78E6C0CB3FCB3FCB3FB4 1520
7 677DE61F6F78E638CB27 1248
8 CB27B56FF1C900000000  976
```

En la Figura 10-9 puede ver el listado completo de esta rutina. La hemos ensamblado a partir de 60500 pero es perfectamente reubicable.

Ahora que podemos calcular la dirección de cada scan, estamos en disposición de preparar las rutinas de scroll vertical. A diferencia con el horizontal, esta vez utilizaremos una sola rutina que será capaz de realizar tanto el scroll ascendente como el descendente, en función del estado de un flag. De nuevo, utilizaremos como flag el bit 0 del registro «C», que deberá estar a «1» para un scroll ascendente y a «0» para uno descendente.

La rutina tendrá dos puntos de entrada: por «SCR-3» realizará un scroll ascendente de un pixel y por «SCR-4» lo hará descendente. Un punto de entrada alternativo sería por «SCR» donde habría que entrar con el flag a «1» y «HL» conteniendo el número 16384 para un scroll ascendente, o bien, con el flag a «0» y «HL» conteniendo 22496 para un scroll descendente.

El procedimiento a seguir consta de tres fases: Primero, transferimos el primer scan o el último (según se trate de scroll ascendente o descendente) al buffer de impresora (dirección 23296); después, entramos en un bucle que va copiando cada scan en el anterior (o posterior) y, finalmente, recuperamos el scan del buffer de impresora y lo colocamos en el último scan (o en el primero).

Como es lógico, el scroll ascendente se realiza de arriba a abajo y el descendente de abajo a arriba, por eso, en el primer caso «HL» debe contener 16384 que es la dirección del primer octeto del primer scan y, en el segundo caso, debe contener 22496 que es la dirección del primer octeto del último scan.

Esta rutina, contendrá parte de la del ejemplo anterior, concretamente, a partir de «SIG-1» y hasta el final excepto el «RET», es decir, desde la línea 140 a la 590 del ejemplo anterior que, esta vez, estarán numeradas desde 250 a 700. La rutina se denomina «SCROLL-VERTICAL» y su listado es el siguiente:

```
100          ORG  55060
110 SCR_3    SET  0,C
120          LD   HL,16384
130          JR   SCR
140 SCR_4    RES  0,C
150          LD   HL,22496
160 SCR      PUSH HL
170          LD   DE,23296
180          PUSH BC
190          LD   BC,32
200          LDIR
210          POP  BC
220          POP  HL
230 BUC_3    PUSH HL
240          PUSH BC
```

```
*HISOFT GENS3M ASSEMBLER*
        ZX SPECTRUM

  Copyright HISOFT 1983
  CURSO C/M MICROHOBBY

Pass 1 errors: 00

            10 *C-
            20 *D+
            30 ;
            40 ;SIG/ANT
            50 ;
60500 100           ORG     60500
60500 110 SIG       SET     0,C
60502 120           JR      SIG_1
60504 130 ANT       RES     0,C
60506 140 SIG_1     LD      A,H
60507 150           AND     #07
60509 160           LD      B,A
60510 170           LD      A,H
60511 180           AND     #18
60513 190           SLA     A
60515 200           SLA     A
60517 210           SLA     A
60519 220           OR      B
60520 230           LD      B,A
60521 240           LD      A,L
60522 250           AND     #E0
60524 260           SRL     A
60526 270           SRL     A
60528 280           OR      B
60529 290           BIT     0,C
60531 300           JR      Z,ANT_1
60533 310           INC     A
60534 320           CP      #C0
60536 330           PUSH    AF
60537 340           JR      SIG_2
60539 350 ANT_1     SUB     1
60541 360           CCF
60542 370           PUSH    AF
60543 380 SIG_2     LD      B,A
60544 390           LD      H,#40
60546 400           AND     #07
60548 410           OR      H
60549 420           LD      H,A
60550 430           LD      A,B
60551 440           AND     #C0
60553 450           SRL     A
60555 460           SRL     A
60557 470           SRL     A
60559 480           OR      H
60560 490           LD      H,A
60561 500           LD      A,L
60562 510           AND     #1F
60564 520           LD      L,A
60565 530           LD      A,B
60566 540           AND     #38
60568 550           SLA     A
60570 560           SLA     A
60572 570           OR      L
60573 580           LD      L,A
60574 590           POP     AF
60575 600           RET

Pass 2 errors: 00

Table used:     69 from     170
```

Fig. 10.9. Listado completo de la rutina «SIG/ANT».

PROGRAMA 1

```
  10 REM     ** SCROLL TOTAL **
  20 REM
  30 REM     Curso C/M MICROHOBBY
  40 REM
  50 CLEAR 54999: LET d=55000
  60 DEF FN a(a$,n)=16*(CODE a$(
n)-48-7*(a$(n)>"9"))+(CODE a$(n+
1)-48-7*(a$(n+1)>"9"))
  65 PRINT #1;"ESPERE 10 SEGUNDO
S"
  70 FOR f=1 TO 18
  80 READ a$,s: LET c=0
  90 FOR n=1 TO LEN a$-1 STEP 2
 100 LET a=FN a(a$,n): POKE d,a
 110 LET d=d+1: LET c=c+a
 120 NEXT n
 130 IF c<>s THEN PRINT "(ERROR)
 en la linea: ";300+10*f: STOP
 140 NEXT f
 200 REM   DEMOSTRACION
 210 CLS : PRINT '''" DEMOSTRACI
ON DE SCROLL TOTAL "'''" PULSE:"
''''" █ Para SCROLL Arriba"''" █
Para SCROLL Abajo"''" █ Para SCR
OLL a Izquierda"''" █ Para SCROL
L a Derecha"
 220 PLOT 0,0: DRAW 239,0: DRAW
0,175: DRAW -239,0: DRAW 0,-175
 230 POKE 23658,0
 240 IF INKEY$="q" THEN RANDOMIZ
E USR 55060
 250 IF INKEY$="a" THEN RANDOMIZ
E USR 55067
 260 IF INKEY$="o" THEN RANDOMIZ
E USR 55000
 270 IF INKEY$="p" THEN RANDOMIZ
E USR 55030
 280 GO TO 230
 300 REM   CODIGO MAQUINA
 310 DATA "21FF570EC00620A7CB16"
,1011
 320 DATA "2B10FB300B22B05CDD2A"
,934
 330 DATA "B05CDDCB20C60D20E8C9"
,1400
 340 DATA "2100400EC00620A7CB1E"
,741
 350 DATA "2310FB300B22B05CDD2A"
,926
 360 DATA "B05CDDCBE0FE0D20E8C9"
,1646
 370 DATA "CBC12100401805CB8121"
,887
 380 DATA "E057E511005BC5012000"
,878
 390 DATA "EDB0C1E1E5C57CE60747"
,1689
 400 DATA "7CE618CB27CB27CB27B0"
,1280
 410 DATA "477DE6E0CB3FCB3FB0CB"
,1561
 420 DATA "4128063CFEC0F51804D6"
,1104
 430 DATA "013FF5472640E607B467"
,1002
 440 DATA "78E6C0CB3FCB3FCB3FB4"
,1520
 450 DATA "677DE61F6F78E638CB27"
,1248
 460 DATA "CB27B56FF1C1D1300BE5"
,1465
 470 DATA "C5012000EDB0C1E118AA"
,1255
 480 DATA "21005B012000EDB0C900"
,771
```

```
===========================
250 SIG_1   ..............
...         ..............
===========================
710         POP  BC
720         POP  DE
730         JR   NC,FIN_1
740         PUSH HL
570         PUSH BC
760         LD   BC,32
770         LDIR
780         POP  BC
790         POP  HL
800         JR   BUC_3
810 FIN_1   LD   HL,23296
820         LD   BC,32
830         LDIR
840         RET
```

Hemos puesto 55060 como dirección de origen para que esta rutina quede situada justo a continuación de las de scroll lateral; pero tanto una como otras son reubicables, así que puede ensamblarlas en cualquier dirección.

Todas las transferencias se realizan con LDIR para que la rutina sea rápida. Como todos los lectores saben, la instrucción «LDIR» exije que «HL» contenga la primera edición del bloque origen, «DE» la del bloque destino y «BC» el número de bytes a transferir; pero, en nuestro caso, estos registros son punteros que contienen las direcciones de inicio de los scans, y el registro «C» contiene el flag que indica si el scroll es ascendente o descendente, así que tendremos que salvar algunos registros en la pila, antes de cada transferencia. Podríamos haber utilizado el set de registros alternativos; pero, en muchos casos, nos valdrá para la transferencia, el valor de algún registro que ya tengamos, valor que perderíamos al ejecutar un «EXX».

Ahora ya, veamos el funcionamiento de la rutina: En 110 y 120 fijamos las condiciones iniciales para un scroll ascendente y saltamos a «SCR» en 130. En 140 y 150 fijamos las condiciones iniciales para un scroll descendente y continuamos en «SCR». Primero, guardamos «HL», cargamos la dirección del buffer de impresora en «DE» (línea 170), guardamos «BC» y cargamos en él la longitud de un scan (línea 190). En 200 transferimos un scan al buffer de impresora y, en 210 y 220, recuperamos los anteriores contenidos de «BC» y «HL». El scan transferido habrá sido el primero si «HL» contenía 16384, o el último si contenía 22496.

A partir de la línea 230, entramos en un bucle donde iremos transfiriendo cada scan al anterior o posterior (en función del estado del flag) y del que saldremos cuando la dirección que nos dé «SIG_1» caiga fuera de la pantalla. Primero, preservamos los contenidos de «HL» y «BC», ya que la rutina «SI_1» los destruye. A continuación, entramos en «SIG_1» que va desde la línea 250 a la 700. En 710 y 720 recuperamos el contenido de «BC» y pasamos a «DE» el anterior contenido de «HL»; el nuevo contenido de «HL» será el scan anterior o el siguiente, es decir, «HL» contendrá el origen y «DE» el destino para efectuar la transferencia de los 32 octetos de un scan. Antes de ello, tenemos que comprobar si estamos, aún, dentro de la pantalla, lo que hacemos en la línea 730 en función del estado del indicador de acarreo (recuerde que, a la salida de «SIG_1», el acarreo está a «1» si la dirección es correcta y a «0» si cae fuera de la pantalla); si el acarreo está a «0», saltamos a «FIN_1», si no, continuamos. En 740 y 750 volvemos a preservar «HL» y «BC» antes de realizar la transferencia. En 760 cargamos la longitud del scan en «BC» y, en 770, transferimos el scan. A continuación, recuperamos «BC» y «HL» y volvemos al inicio del bucle. En este punto, «HL» contiene la dirección del siguiente scan a procesar, es decir, el scan de destino de la siguiente transferencia; de esta forma, en sucesivas pasa-

MICRO-1

el IVA lo paga MICRO-1

C/ Duque de Sesto, 50. 28009 Madrid
Tel.: (91) 275 96 16/274 53 80
(Metro O'Donell o Goya)
Aparcamiento gratuito en Felipe II

SOFTWARE: ¡¡2 PROGRAMAS POR EL PRECIO DE 1!!
Y además, completamente gratis, un magnífico reloj de cuarzo. Increíble ¿verdad?

	Ptas.
TURBO ESPRIT	2.100
DYNAMITE DAN	2.100
THEY SOLD A MILLION	2.500
FIGHTER PILOT	1.975
MASTER OF T. LAMP	1.950
NIGHIT SHADE	1.950
HACKER	1.950
SUPER TEST	2.300
KNIGHT LORE	1.750
ALIEN 8	1.750
TOMAHAWK	2.300
THREE W. PARADISE	2.100
COSMIC WARTOAD	2.100
LEYENDA DE AMAZONAS	2.300

	Ptas.
PING PONG	2.295
SABOTEUR	2.295
RAMBO	2.295
YIEAR KUNG FU	2.295
WORLD SERIES BASEBALL	2.095
MAPGAME	2.750
RAID	2.295
HYPERSPORTS	2.295
HIGHWAY ENCOUNTER	1.750
NGHT SHADE	1.750
GUNFRIGHT	1.950
BATALLA DE LOS PLANETAS	2.100
MOVIE	2.100
N.O.M.A.D.	2.100

Software de regalo (Oferta 2 x 1): Shadow Fire, Dragontorc, Dummy Run, Fighting Warrior, Psi Warrior, Southern Belle, Poole position, Mapsnatch.

SPECTRUM PLUS+6 JUEGOS
¡¡27.800 ptas.!!
Gratis: 2 Walkie Talkies

NUEVOS PROGRAMAS EXPLOSIVOS
Ali Bebe, Kripton Raiders, Tommy. 1895 ptas. los tres
y además gratis 1 calculadora

Lápiz óptico DK'Tronics 2.890 ptas.

Interface 1 10.900 ptas.
Microdrive 10.900 ptas.

Cassette especial para ordenador 5.295 ptas.

Ampliaciones de memoria
¡¡3.995 ptas.!!

Teclados profesionales
Saga 1 9.295 ptas.
Indescomp 13.195 ptas

Wafadrive 16.995 ptas.
Quick disk 2.8'' 23.995 ptas.

OFERTA IMPRESORAS: TODAS LAS MARCAS
¡¡20% DTO. SOBRE P.V.P.!!

Precios excepcionales para tu Amstrad CPC-464, CPC-6128 y PCW-8256

Ofertas: Joystick Quick Shot
Quick Shot II+Interface 3.695 ptas.
Quick Shot V+Interface 3.995 ptas.

UNA CALCULADORA GRATIS

Servicio técnico de reparación
Tarifa fija: 3.600 ptas.

Cartuchos Microdrive	495
Diskettes 5¼''	295
Diskettes 3''	990
Cartucheras para Microdrive	150

Cinta C-15 especial ordenador	69
Interface Centronics/RS-232	8.495
Amplificador de sonido	2.390
Interface doble Kempston	2.795

Pedidos contra reembolso sin ningún gasto de envío. Tels. (91) 275 96 16/274 53 80, o escribiendo a Micro-1. C/ Duque de Sesto, 50. 28009 Madrid

Descubre los secretos de los juegos comerciales

EL ESPIA

Francisco ALEJANDRE

La mayoría de usuarios de microordenadores, aunque en un principio sólo se dedican a jugar, tarde o temprano experimentan la necesidad de crear sus propios programas y para ello nada mejor que «espiar» a los profesionales. Pensando en todos ellos hemos elaborado este utilísimo programa que permite investigar la estructura gráfica de cualquier juego.



El programa responde principalmente a una idea: satisfacer la curiosidad de todos aquellos que sienten gran pasión por la programación o simplemente por dibujar, y desean averiguar los secretos mejor guardados de los programas comerciales: cómo están almacenados en la memoria los distintos gráficos de los personajes, escenarios y situaciones.

Si os preguntáis para qué sirve, os diremos que depende. Se puede utilizar simplemente para pasar el tiempo viendo todos los gráficos de un juego o puede servir para algo más interesante como coleccionarlos o incluso animarlos.

De la misma forma que se coleccionan pantallas se pueden coleccionar gráficos. Además, es un programa constructivo pues al ver juegos como el Three Weeks is Paradise o incluso el propio Alien 8, se observa que los gráficos tienen al lado una especie de silueta que sirve para que un gráfico se superponga en otro sin mezclarse (mediante operaciones AND y OR) y esto es la base del sistema FILMATION, del que pronto hablaremos.

En fin, vosotros mismos podéis juzgarlo. En nuestra opinión es bastante útil, y si nos permitís un consejo, utilizarlo con Joystick Kempston, pues resulta infinitamente más cómodo.

El programa Basic (programa 1) debéis teclearlo y salvarlo con autoejecución. De esta forma cargará automáticamente los dos bloques de bytes (listados 2 y 3) que previamente habremos grabado en cinta utilizando el Cargador Universal de Código Máquina, siguiendo el siguiente método: primero se teclea el listado 2 y se efectúa un DUMP indicando como dirección la 40000 y como número de bytes 142. Luego se salva en cinta a continuación del programa Basic. Por último se teclea el listado 3 y se realiza un DUMP en la 40000, indicando 64 como número de bytes. Habrá de grabarse a continuación del segundo.

Manejo y funcionamiento

El programa comienza a funcionar, una vez cargados los tres bloques y lo primero que pregunta es si se va a utilizar joystick o teclado. Una vez cargado el bloque de gráficos que queremos inspeccionar siguiendo los pasos que más adelante se explican y suponiendo que hemos elegido joystick, el manejo sería el siguiente:

—ARRIBA: se acelera el paso de gráficos hacia arriba.

—ABAJO: se frena el paso de gráficos.

—IZQUDA: disminuye columnas.

—DCHA: aumenta columnas.

Su utililidad puede ser doble: por un lado sirve para ver cómo están almacenados los gráficos en los juegos comerciales y si queremos, incluso coleccionarlos nosotros con la opción SAVE.

Por otro lado, puede servirnos en la realización de nuestros propios programas en cuanto que podemos ver sin dificultad las direcciones donde hemos ido almacenando nuestros gráficos y poder hacer así las modificaciones pertinentes.

El programa en sí se apoya en la rutina de impresión en CM que necesita la dirección y las columnas del gráfico. Las filas son siempre 14 que es el lado cuadrado donde se visualizan por pantalla.

Habiendo cargado con la opción LOAD el bloque principal de un juego lo que hace el programa es ir sacando por pantalla sucesivos gráficos de tantas columnas como nosotros se lo indiquemos moviendo el joystick o pulsando «o» o «p». Es decir, al avanzar, lo que hace es ir pokeando direcciones su-



cesivas en la rutina de impresión, lo cual da la sensación de scroll en la pantalla. Según como lo veamos podemos ir ajustando las columnas con las mencionadas teclas.

Si los gráficos que van pasando se ven mal, podemos cambiar el contraste de la pantalla con la opción COLOR o centrarlos si están descolocados, etc...

Estructura

El programa se compone de 3 partes, que son:

- un bloque en el buffer de impresora.
- un bloque de 8 gráficos definidos.
- el programa principal.

PROGRAMA 1

```
   1 REM ALEJANDRE '86
   2 BORDER 0: PAPER 0: BRIGHT 1: INK 7: CLEAR 27935
   3 LOAD ""CODE 23296: LOAD ""CODE 27936
   4 POKE 23675,33: POKE 23676,109: POKE 23658,8: GO SUB 9000: INK 5
   5 LET H=0: LET R=1
  10 LET DIR=28000: LET COL=1: LET PAS=COL: GO SUB 400
  20 INPUT "(1) TECLADO O (2) KEMPSTON ";key
  30 IF key<>1 AND key<>2 THEN GO TO 20
  50 REM BUCLE
  60 RANDOMIZE DIR: POKE 23297,PEEK 23670: POKE 23298,PEEK 23671: POKE 23313,COL
  70 RANDOMIZE USR 23296
  75 PRINT INK 7;AT 1,23;dir
  80 IF key=1 THEN GO SUB 9500: GO TO 82
  81 LET a=IN 223
  82 IF a=2 THEN : LET col=col-1: GO SUB 8000: LET pas=col
  83 IF a=1 THEN LET col=col+1: LET pas=col
  84 IF a=17 THEN LET dir=dir-1
  85 IF a=18 THEN LET dir=dir+1
  86 IF a=4 AND R=0 THEN LET DIR=DIR-COL
  87 IF a=8 AND R=0 THEN LET DIR=DIR+COL
  88 IF COL>14 THEN LET COL=14
  89 IF COL<1 THEN LET COL=1
  95 IF a=20 THEN GO TO 460
  97 IF A=24 THEN LET R=0: GO TO 60
  99 IF R=0 THEN GO TO 60
 100 IF a=4 THEN LET pas=pas-col
 101 IF a=8 THEN LET pas=pas+col
 110 LET DIR=DIR+PAS
 115 IF DIR>65535 THEN LET DIR=28000
 116 IF DIR<28000 THEN LET DIR=65535
 120 GO TO 60
 390 REM SCREEN
 400 PRINT AT 0,3;"ABBBBBBBBBBBBBBC": FOR i=1 TO 14: PRINT AT i,3;"G              H": NEXT i: PRINT AT 15,3;"DEEEEEEEEEEEEEEF"
 410 PRINT AT 0,22;"ABBBBBC": PRINT AT 1,22;"G     H": PRINT AT 2,22;"DEEEEEF": PRINT AT 4,22;"ABBBBBC": FOR i=5 TO 13: PRINT AT i,22;"G     H": NEXT i: PRINT AT 14,22;"DEEEEEF"
 414 PRINT AT 18,3;"ABBBBBBBBBBBBBBC": PRINT AT 19,3;"G              H": PRINT AT 20,3;"DEEEEEEEEEEEEEEF"
 415 INK 7: PRINT AT 5,23;"Start";AT 7,23;"Color";AT 9,23;"Point";AT 11,23;"Load.";AT 13,23;"Save."
 420 RANDOMIZE USR 23343: RETURN
 455 REM MOV.JOY
 460 LET x=22711: LET k=0
 470 LET x1=x
 480 IF key=1 THEN GO SUB 9500: GO TO 510
 500 LET a=IN 223
 510 BEEP .0007,60
 520 LET x=x1+64*(a=4)-64*(a=8)
 525 IF a=16 THEN GO TO 600
 530 IF x<22711 THEN LET x=22711: GO TO 540
 535 IF x>22967 THEN LET x=22967: GO TO 540
 537 LET k=k+(a=4)-(a=8)
 540 FOR i=x1 TO x1+4: POKE i,71: NEXT i: FOR i=x TO x+4: POKE i,66: NEXT i: GO TO 470
 600 BEEP .1,30: IF K=0 THEN LET R=1: LET PAS=COL: GO TO 50
 605 REM MENU
 610 GO SUB 1000*K
 650 GO TO 500
1000 LET S=23353: IF PEEK S=70 THEN POKE S,112: GO TO 1200
1100 POKE S,70
1200 RANDOMIZE USR 23343: PAUSE 10: RETURN
2000 PRINT AT 19,5;"28000  46000": PRINT INK 2; BRIGHT 0; FLASH 1;AT 19,4;"█": LET F=2
2090 IF key=1 THEN GO SUB 9500: GO TO 2200
2100 LET a=IN 223
2200 IF a=2 THEN LET F=2: PRINT FLASH 1; INK 2; BRIGHT 0;AT 19,4;"█";AT 19,17; FLASH 0;" "
2300 IF a=1 THEN LET F=1: PRINT AT 19,17; FLASH 1; BRIGHT 0; INK 2;"█";AT 19,4; FLASH 0;" "
2400 IF a=4 THEN IF F=2 THEN LET DIR=28000: GO TO 2600
2450 IF a=4 THEN IF F=1 THEN LET DIR=46000: GO TO 2600
2500 GO TO 2090
2600 PRINT AT 19,4; FLASH 0;"              ": RETURN
3000 RANDOMIZE USR 23367: RETURN
4000 INPUT "Longitud?";lo
4001 INPUT "Nombre?";s$: IF LEN s$=0 OR LEN s$>10 THEN GO TO 4001
4002 SAVE s$CODE dir,lo: RETURN
8000 LET w=16420+col: LET ww=18436+col
8100 RANDOMIZE w: POKE 23385,PEEK 23670: POKE 23386,PEEK 23671: RANDOMIZE ww: POKE 23422,PEEK 23670: POKE 23423,PEEK 23671: RANDOMIZE USR 23381
8200 RETURN
9000 INK 6: BRIGHT 0: PRINT INVERSE 1;AT 18,21;"MICROHOBBY";AT 20,21;" EL ESPIA ": BRIGHT 1: RETURN
9500 LET a$=INKEY$
9510 LET a=16*(a$="0")+2*(a$="O")+(a$="P")+4*(a$="A")+8*(a$="Q")+17*(a$="8")+18*(a$="5")+20*(a$="6")+24*(a$="7"): RETURN
```

a) Bloque de impresora: ocupa las direcciones 23296 hasta 23428.

Hay en él 4 rutinas a saber:

— 23296: rutina de IMPRESION principal que se encarga de la salida por pantalla y a la que se accede pokeando dirección y columnas.

— 23343: rutina de COLOR del cuadrado, a la que se accede pokeando el atributo correspondiente.

— 23367: rutina de CARGA.

— 23384: rutina de impresión semejante a la primera, pero que se encarga del borrado de la columna correspondiente cuando éstas son disminuidas.

b) Gráficos definidos: deben ubicarse en la dirección 27936 y sirven para la realización del marco de la pantalla.

c) Programa principal: veámoslo por bloques de líneas:

3-30 Inicialización: se baja la Ramtop a la 27935, se pokean las nuevas direcciones de los gráficos definidos, se pone en modo mayúsculas y se dibuja

Para la opción TECLADO, las teclas son como siguen:

Tecla		Función
P	→	Teclas de movimiento normal, necesarias tanto para el menú como para la rutina POINT.
O	←	
Q	↑	
A	↓	
5	→	Teclas accesorias que simulan los movimientos de joystick con disparo pulsado.
6	↓	
7	↑	
8	←	
0 Fuego		Para relacionar el menú.

LISTADO 2 - Buffer de impresora

```
1   1187FF2124400602C506  751
2   07C50608C5E506011A77  796
3   231310FAE101000109C1  749
4   10EE01E007ED42C110E3  1225
5   210448C110DAC9212458  894
6   060EC5E5060E3E467723  752
7   10FAE10120000 9C110EE  980
8   C9373EFFDD21606D1170  1161
9   94CD5605C90000002127  717
10  400602C50607C50608C5  690
11  E5060136002313 10FAE1  835
12  01000109C110EE01E007  690
13  ED42C110E3210748C110  1060
14  DAC90000000000000000  419
```

LISTADO 3 - (U.D.G.)

```
1   3E001F0F476371787C00  635
2   FFFFFFFFFFFF000000F8F0  1763
3   E2C68E1E3E7C78716347  1185
4   0F1F000000FFFFFFFFFF  1321
5   003E1E8EC6E2F0F8007C  1270
6   7C7C7C7C7C7C7C3E3E3E  1054
7   3E3E3E3E000000000000  248
```

DESENSAMBLE DE LA RUTINA

```
10          ORG  23296
20          LD   DE,65415
30          LD   HL,16420
40          LD   B,#02
50 LABEL0   PUSH BC
60          LD   B,#07
70 LABEL1   PUSH BC
80          LD   B,#08
90 LABEL2   PUSH BC
100         PUSH HL
110         LD   B,#01
120 LABEL3  LD   A,(DE)
130         LD   (HL),A
140         INC  HL
150         INC  DE
160         DJNZ LABEL3
170         POP  HL
180         LD   BC,00256
190         ADD  HL,BC
200         POP  BC
210         DJNZ LABEL2
220         LD   BC,02016
230         SBC  HL,BC
240         POP  BC
250         DJNZ LABEL1
260         LD   HL,18436
270         POP  BC
280         DJNZ LABEL0
290         RET
300         LD   HL,22564
310         LD   B,#0E
320 LABEL4  PUSH BC
330         PUSH HL
340         LD   B,#0E
350 LABEL5  LD   A,#46
360         LD   (HL),A
370         INC  HL
380         DJNZ LABEL5
390         POP  HL
400         LD   BC,00032
410         ADD  HL,BC
420         POP  BC
430         DJNZ LABEL4
440         RET
450         SCF
460         LD   A,#FF
470         LD   IX,28000
480         LD   DE,38000
490         CALL 01366
500         RET
510         NOP
520         NOP
530         NOP
540         LD   HL,16423
550         LD   B,#02
560 LABEL6  PUSH BC
570         LD   B,#07
580 LABEL7  PUSH BC
590         LD   B,#08
600 LABEL8  PUSH BC
610         PUSH HL
620         LD   B,#01
630 LABEL9  LD   (HL),#00
640         INC  HL
650         INC  DE
660         DJNZ LABEL9
670         POP  HL
680         LD   BC,00256
690         ADD  HL,BC
700         POP  BC
710         DJNZ LABEL8
720         LD   BC,02016
730         SBC  HL,BC
740         POP  BC
750         DJNZ LABEL7
760         LD   HL,18439
770         POP  BC
780         DJNZ LABEL6
790         RET
```

la pantalla, preguntando después por la opción:

1.Teclado
2.Kempston

Según cual elijamos la variable KEY valdrá 1 ó 2. Durante la ejecución, cuando valga 1 se hará una llamada a la 95ØØ, donde está una subrutina que devuelve en «a» el valor de la tecla pulsada con el fin de hacerlo compatible con los controles del joystick.

5Ø-12Ø Bucle principal en el que se van pokeando sucesivas direcciones en la rutina de impresión y otros valores de acuerdo con nuestros movimientos.

Aquí, como en otras partes del programa, si KEY vale 1 se hace caso omiso del IN 223.

39Ø-42Ø Dibujo de la pantalla.

455-6ØØ Se controla el movimiento en el menú de opciones con teclado o joystick.

6Ø5-65Ø Se llama a la subrutina que corresponda a la opción seleccionada.

1ØØØ-12ØØ Subrutina del color que cambia el contraste de la pantalla pokeando el atributo correspondiente en la propia rutina.

2ØØØ-26ØØ Subrutina POINT que se encarga de alterar la variable DIR para pasar rápidamente de una parte a otra de la memoria.

3ØØØ Se llama a la rutina de carga.

4ØØØ Se piden los datos necesarios para salvar un determinado bloque.

8ØØØ-82ØØ Calculan la columna que debe ser borrada si estamos pulsando «O» y llama a la rutina en CM.

95ØØ Rutina de control del teclado.

Como podréis observar, se ha bajado el CLEAR a la 27935 y situado los gráficos en la 27936 con objeto de reservar la mayor cantidad de memoria posible para el bloque de bytes a inspeccionar, que de esta forma puede ocupar desde la dirección 28ØØØ hasta la 65535, ambas inclusive.

Por ejemplo, si estamos con joystick y queremos centrar un gráfico hemos de dar a izquierda o derecha con el disparo pulsado. Si estamos con teclado, no es necesario ya que ambas teclas están unidas en el 5 (disparo + izq.) y el 8 (disparo + dcha.). Lo mismo sucede con las otras:

— Con el 6 se para el menú.
— Con el 7 se pone pixel a pixel luego habrá que mover con «q» y «a».

Esperemos que hagais un uso adecuado de esta información tan «confidencial».

¡GRATIS! (Solo hasta el 31 de Mayo)

2 cintas. Los 20 mejores programas de utilidades al suscribirte a: MICROHOBBY SEMANAL

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

PROGRAMACION

Ensamblador Confección de programas en código máquina, mediante la utilización de mnemónicos del lenguaje Assembler del Z80.

Desensamblador Para obtener el listado de un programa en código máquina de forma similar al de los programas Basic.

Cargador de C/M Os ayudará introducir sin error rutinas en código máquina.

Depurador Ejecución a «cámara lenta» de cualquier programa Basic. Extremadamente útil para depurar programas.

Monitor Este curioso programa permitirá corregir cualquier texto de otro en C/M.

RUTINAS

Cirujano Rutina para reparar programas cargados defectuosamente.

Tokens Localización de Tokens (palabra clave en el Spectrum).

Microcat Los datos relativos a la información guardada en un cartucho de Microdrive son de vital importancia.

Renumerador Uno de los comandos de edición más potentes y apreciados por los programadores.

Conversor Números decimales a hexadecimales y viceversa en cualquier momento.



COMPLEMENTOS

Listador Todos los datos relativos al contenido de cada cinta de cassette al descubierto.

Analizador de Audio Trazado gráfico de señales de audio para grabaciones en cassette.

Agenda Fichero personal de direcciones.

Copyupi Copiador de programas de cinta a cinta para efectuar copias de seguridad con toda fiabilidad.

Editext Procesador de textos a 64 columnas.

GRAFICOS Y SONIDO

Screens Editor de gráficos para creación de espectaculares pantallas.

Multi U.D.G. Editor de gráficos definidos de gran capacidad y versatilidad.

Traspa Copiador de caracteres, gráficos y atributos.

Databeep La ejecución de melodías en CM ofrece las ventajas de mejor ocupación y mayor calidad musical.

Voz Sintetizador de voz.

Recorta o copia el cupón que va cosido en las páginas de esta revista.

(Válido para España y Portugal)

LOS JUSTICIEROS DEL SOFTWARE

LA LEYENDA DE LAS AMAZONAS

Entre leyendas y mujeres bélicas, se desarrolla este juego de US GOLD que esta semana sentamos en el banquillo. Nuestros justicieros se encargarán de decir sobre él la última palabra.

«Poco original»

● **Positivo**

El movimiento está muy bien conseguido, así como el scroll de la pantalla. El personaje responde a sus movimientos tanto por el teclado como por el joystick. Tiene una buena presentación y detalles gráficos. La dificultad no es muy alta con lo cual hace que este juego sea bastante adictivo.

● **Negativo**

No es original tanto por su tema como por el desarrollo del mismo. Se echa en falta los efectos sonoros y el colorido.

Puntuación: 7

Ricardo Alonso

«Buena presentación»

● **Positivo**

Los movimientos de los personajes y sus excelentes gráficos son lo más positivo de este juego. Hay que destacar también la presentación del mismo y el scroll de la pantalla. Es un programa muy adictivo aunque tenga algo de dificultad.

● **Negativo**

Se echa en falta algunos efectos sonoros y de colorido que podríamos esperar de este tema.

Puntuación: 7

José Antonio Galiana

«Algo lento»

● **Positivo**

Los gráficos son muy buenos y el movimiento de los personajes y objetos, junto con el suave scroll de la pantalla, hace que parezca muy real. Efecto de fondo muy logrado.

● **Negativo**

Es algo lento y hay pocos efectos sonoros.

Puntuación: 8

Gabriel Marti

«Un juego muy adictivo

● **Positivo**

A pesar de la cantidad de teclas que hay que manejar, no se hace dificultoso, requiriéndose, eso sí, un poco de práctica. De dificultad mediana, se hace un juego muy adictivo. Los gráficos son excelentes y el movimiento de los personajes están muy logrados.

● **Negativo**

La falta de colorido en los personajes diferenciándolos del resto de la escena,

Puntuación: 8

Fco. Javier Cano

«El movimiento, lo mejor»

● **Positivo**

El movimiento es lo mejor del juego. El jugador puede elegir el tipo de golpe que desee. El control por teclado está muy bien pensado. Los gráficos están bastante conseguidos, pero quizá un fallo sea el no poder distraerse para observar el paisaje.

● **Negativo**

El sonido es bastante flojo y no estaría mal que en pantalla apareciera un indicador con la inteligencia de las amazonas.

Puntuación: 8

Pilar Arias

«Sonido precario»

● **Positivo**

Cabe destacar del programa de U. S. Gold sus movimientos y sus excelentes gráficos. Es un gran programa de arcade muy adictivo.

● **Negativo**

No es demasiado original y no corresponde a un programa de última hora. El sonido es algo precario.

Puntuación: 8

Juan Markessinis

«Poca respuesta al teclado»

● **Positivo**

Gráficos con gran calidad. El movimiento de los personajes es muy bueno y el scroll de la pantalla muy suave.

● **Negativo**

No responde mucho al teclado. No es muy adictivo y no tiene apenas efectos sonoros.

Puntuación: 8

Roberto Carlos Alonso

«Manejo monótono»

● **Positivo**

Los gráficos de fondo y los personajes, así como sus movimientos son muy parecidos a los del Tir Na Nog, lo cual supone algo más a su favor. El scroll del fondo de pantalla es muy suave. Resulta agradable a la vista. Un factor muy positivo incluido en este juego es el tiempo.

● **Negativo**

Las teclas tardan en responder y esto, unido a que hay que usar bastantes, hace que su manejo sea muy complicado.

Puntuación: 7

David de Llodio



EL PRIMER VEREDICTO

Han transcurrido ya más de cinco meses desde que comenzamos esta serie que dimos en llamar «Los Justicieros del Software». Con ella ponemos semanalmente nuestra revista a vuestra disposición para que podáis exponer vuestras opiniones a cerca de aquellos programas que, por una razón u otra, adquieren una mayor relevancia.

Durante estas 22 semanas de existencia hemos, o mejor dicho, habéis comentado y otorgado vuestras objetivas puntuaciones a un total de 17 juegos pertenecientes a otras 15 casas de software diferentes.

Y ya ha llegado el momento de hacer una breve pausa y recapacitar sobre todo lo ocurrido hasta el momento en este largo camino que ha de llevarnos, dentro de unos meses, a la elección del programa que conseguirá el codiciadísimo galardón de **Programa del Año**.

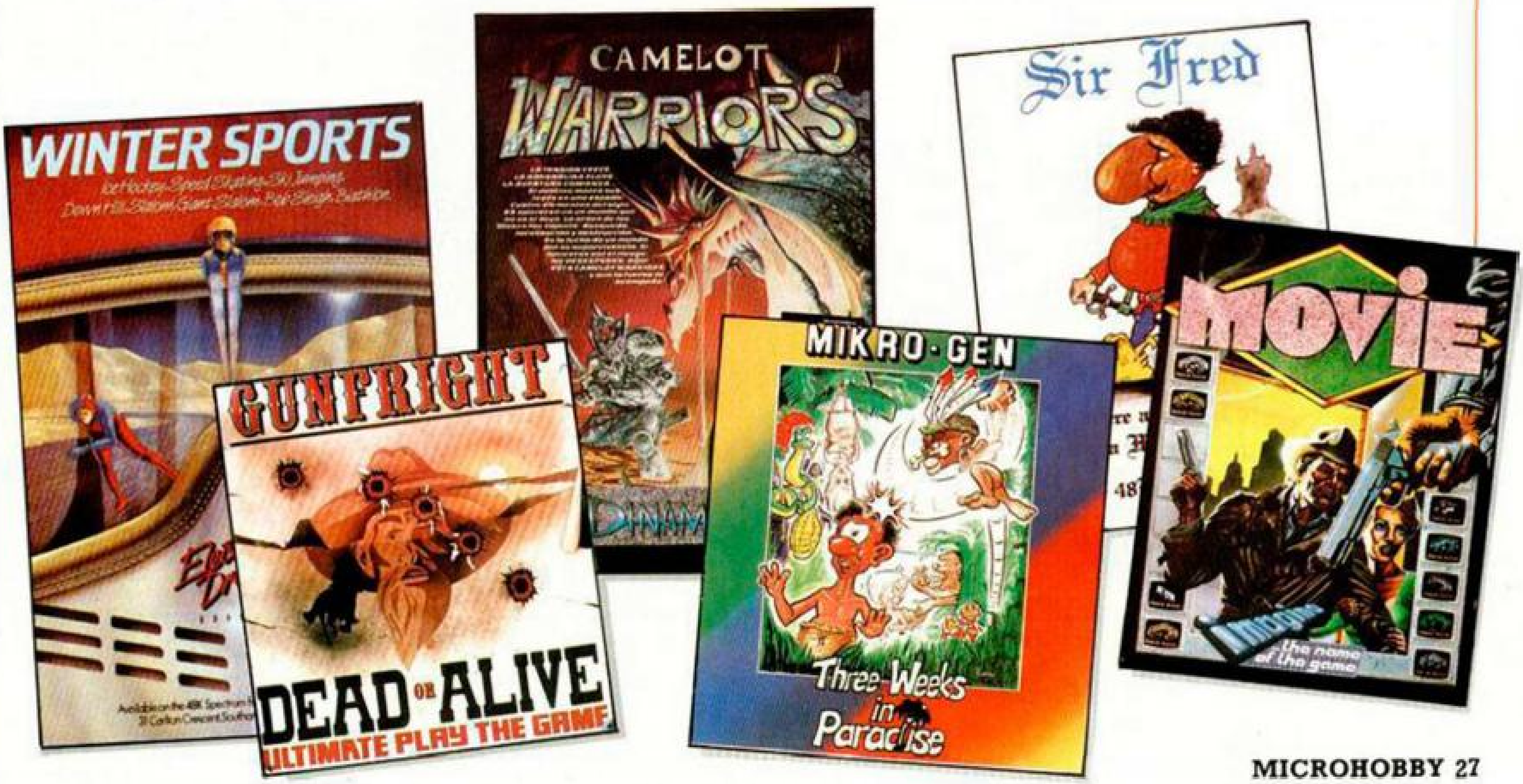
A continuación, os presentamos la lista completa que incluye todos los programas que hasta el momento habéis comentado a través de estos expertos justicieros, así como la casa a la que pertenecen y la calificación media obtenida.

Esta relación habla por sí sola, pero merece destacar un hecho muy importante. De entre los seis primeros puestos, tres de ellos están ocupados por programas españoles: Camelot Warriors y Olé Toro, de Dinamic y Sir Fred, de Made in Spain. Esto, evidentemente, demuestra la gran calidad que el software está alcanzando en nuestro país, el cual puede equipararse a los más prestigiosos programas ingleses y americanos.

Hasta la fecha, la primera posición está ocupada (y posiblemente va a ser muy difícil que sea desbancado) por el sensacional Movie, mientras que también llama la atención la baja puntuación conseguida por otro juego que, a priori, parecía uno de los más firmes candidatos al título: Fairlight.

De todas formas aún faltan muchos meses para que acabe la competición y en cualquier momento puede aparecer la sorpresa.

1. MOVIE	Imagine	9,40	9. YABBA DABBA DOO	Quicksilva	7,60
2. CAMELOT WARRIORS	Dinamic	9,00	10. WEST BANK	Dinamic	7,50
3. THREE WEEKS IN PARADISE	Mikro-Gen	8,80	11. FAIRLIHGT	The Edge	7,50
4. OLE TORO	Dinamic	8,50	12. I, OF THE MASK	Electric Dreams	7,20
5. GUNFRIGHT	Ultimate	8,50	13. MICKIE	Imagine	6,80
6. SIR FRED	Made in Spain	8,40	14. WINTER SPORTS	Electric Dreams	6,80
7. SABOTEUR	Durell	8,25	15. FIGHTING WARRIOR	Melbourne House	6,75
8. AMAZON WOMEN	Silvertime	7,65	16. MARSPORT	Gargoyle Games	6,70
			17. POPEYE	Dk'Tronics	6,50



Rutinas de carga en Código Máquina

LA BIBLIA DEL «HACKER» (VII)

José Manuel LAZO

Ya es hora de que estudiemos las distintas maneras en que puede cargarse un programa desde CM. En primer lugar veremos la correcta utilización de la rutina LOAD de la ROM.

Partimos del supuesto de que habéis aprendido ya los fundamentos que se han sentado en los capítulos anteriores sobre protecciones a nivel Basic, aunque volveremos a ello después, cuando nos centremos en la protección «turbo».

Ahora vamos a introducirnos de lleno ya en lo que se puede llamar protecciones a nivel CM, esto es, cuando el cargador del programa ejecuta una llamada a una rutina en CM para seguir cargando el resto del mismo.

La estructura general del cargador CM puede ser ésta:

```
LD A,255
LD IX,16384
LD DE,6912
SCF
CALL#556
LD A,255
LD IX,25000
LD DE,40000
SCF
CALL#556
......
```

Una asignación de vectores y unas llamadas a rutinas de la ROM. Este es el caso más sencillo que usa la rutina de la ROM LOAD ubicada en la dirección #556 (1366 en decimal).

La rutina LOAD

Es muy interesante antes de proseguir, echar un vistazo al funcionamiento de la rutina LOAD de la ROM. Si de todas formas deseáis profundizar más sobre el tema os podéis dirigir al Especial n.º 2 de MICROHOBBY, donde se trata con mayor detalle este tema.

Esta rutina utiliza el registro IX para contener la dirección de comienzo donde se van a cargar los bytes, el registro DE para contener la longitud del bloque y el registro A para el flag de identificación.

Pero ¡ojo!, esto carga sin la «cabecera» que contiene la información del nombre y longitud de los bytes, lo cual trae consigo el que se cargue lo primero que entre.

Si observáis el *Gráfico 1* podréis ver la manera en que están grabados unos bytes o un programa en la cinta: en primer lugar, el tono guía, y luego, la cabecera en sí que contiene un primer byte como flag de identificación (0) y otros **16** con la información de cabecera: nombre, comienzo, longitud, tipo y demás...

El segundo bloque es el que os interesa, es lo que se llama: «carga sin cabecera» ya que se prescinde de la misma, de lo cual se deduce que debemos de dar los valores de la dirección y longitud del bloque de datos en los registros que arriba se exponen.

Al elevar el banderín de Carry con la instrucción SCF provocamos que la rutina de la ROM cargue, ya que de lo contrario, sólo verificaría.

Primeros trucos en Assembler

Esta es una estructura general suponiendo que el programa al cargarse no tuviera cabecera y entrase a velocidad NORMAL. Por regla general se ha de buscar una asignación de vectores en los registros IX y DE los cuales indican comienzo y longitud, unas llamadas a rutinas cargadoras y un retorno a Basic o un salto ya al programa en sí.

Pero hay muchas formas de enredar esto tan sencillo para hacerlo menos inteligible.

Sentemos primero unos sencillos conceptos de Assembler:

En primer lugar la instrucción CALL dirección significa, como todos vosotros sabéis, una llamada a una rutina en CM., pero agrupa una serie de operaciones como son:

CALL dirección = PUSH PC (Program Counter) JP dirección.

En segundo lugar, la instrucción RET que sirve para retornar de una rutina CM. Tendría el siguiente significado, en nuestros mnemónicos imaginarios:

RET = POP PC o JP (pila).

De esto, se deduce que cuando efectuamos un CALL guardamos la dirección de retorno en la pila, y si efectuáramos otro se guardaría la nueva encima sin borrarse la antigua de forma que los RET que se vayan ejecutando van sacando esas direcciones de retorno de la pila.

Es muy sencillo pokear la dirección de retorno en la rutina cargadora y cambiarla por otra para que la instrucción RET del final no ejecute un retorno a Basic como sería de esperar, sino un salto directo al programa en CM.

Por ejemplo:

```
LD IX,25000
LD DE,1000
LD A,255
SCF
CALL#556
POP HL
LD HL,25000
PUSH HL
RET
```

Esto sería un ejemplo de una rutina que cargase otra en la dirección 25000 y a continuación ejecutase una llamada a esta rutina con la instrucción RET, fijaos bien en su estructura pues abunda más de lo que sería de esperar.

Otra forma es terminar el programa en vez de con un RET, con un JP a la rutina LOAD de la ROM, ya que el RET se halla en la propia rutina de la ROM.

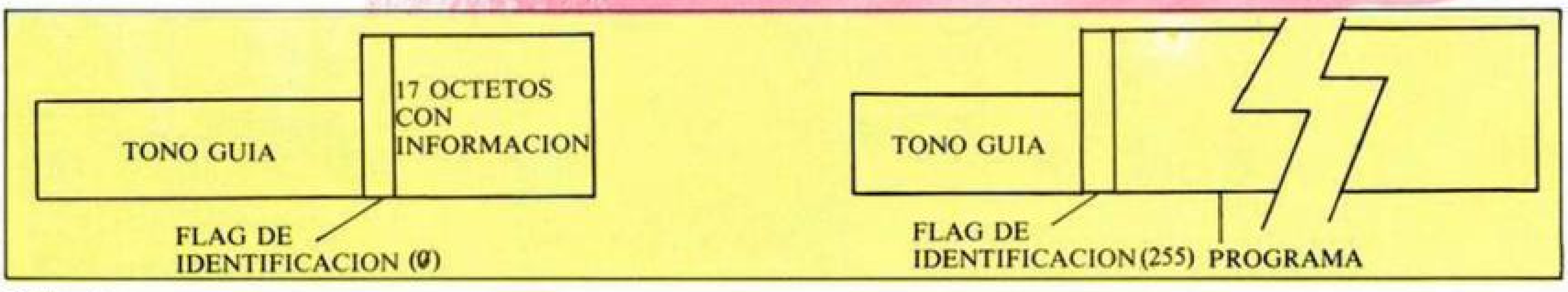


Gráfico 1.

MICRO Manía
Sólo para adictos

MICKIE

Tomás • Madrid • Mickie • Trucos • Amigos MICROHOBBY • Gracias • De nada •



1.° En la primera clase, justo en el medio de la mesa del profesor, en el libro, dando tres gritos saldrá una caricatura que nos dará 1.000 puntos de premio.

2.° Cuando salimos de la primera clase, justo al lado de la puerta hay un diploma que dando gritos saldrán 1.000 puntos de premio.

3.° Cuando entramos en la segunda clase, en medio de las dos puertas, hay una especie de urnas, en la segunda empezando por la puerta de la izquierda dando unos gritos saldrá una caricatura y 1.000 puntos de premio.

4.° Cuando nos disponemos a entrar en la tercera clase, cuando terminamos de bajar las escaleras hay una especie de cuadro con dos torres, si nos ponemos en el medio y damos gritos saldrá la palabra «JOF» y nos dará 1.000 puntos de premio.

5.° Cuando entremos en la tercera clase y si tenemos un poco de tiempo en el horno en una especie de rectángulo (de los tres el del medio) dando gritos saldrá una caricatura con 1.000 puntos de premio.

6.° Cuando salimos de la tercera clase y nos dirigimos a la cuarta clase, cuando terminamos de bajar las escaleras hay un cuadro con dos torres, si nos ponemos en medio y damos gritos saldrá de nuevo la palabra «JOF» y 1.000 puntos de premio.

7.° Cuando entramos en el gimnasio de las chicas, en el altavoz izquierdo dando gritos saldrá una especie de caricatura y 1.000 puntos de premio.

8.° Cuando salimos del gimnasio de las chicas entre dos puertas hay un diploma que dando tres gritos saldrá una caricatura y 1.000 puntos de premio.

9.° En todas las clases los corazones se van iluminando sucesivamente, si los echamos cuando está iluminado, cuando terminamos de echarlos, si lo hemos hecho bien, saldrá la palabra «PERFECT» y nos dará unos puntos.

¿Qué? No está mal, ¿no? Vamos, que si no conseguís batir el récord mundial de puntuación, cogemos y cerramos esta sección.

TOMMY

Para completar la información correspondiente al día de hoy, tenemos el gusto de ofreceros dos excelentes Pokes para el no menos maravilloso Tommy.

POKE 38941, 0 si se desean vidas infinitas.

POKE 38941, 60 si se desea que al chocar con algún objeto la cantidad de dinero aumente en lugar de disminuir. Además, se conseguirá la módica cantidad de 255 vidas.

Si pasáis por La Coruña darle las gracias a Fernando Mouriño de nuestra parte.



LIBROS



MINI DICCIONARIO INFORMATICO

R. Tapias
Ed. Noray. 192 páginas

Todos los que algún día decidimos adentrarnos en este apasionante tema de la informática, nos encontramos, tarde o temprano con palabras absolutamente desconocidas, además de otras que nos eran familiares pero eran utilizadas fuera de su contexto habitual.

Este pequeño diccionario de bolsillo tiene la misión de descubrirnos, de manera clara y concisa, pero con un grado de técnica suficiente, los ocultos significados de esas extrañas palabras que pronto se nos convertirán en familiares.

Aunque la mayoría de los vocablos utilizados en informática proceden del idioma inglés, el diccionario se ha confeccionado por orden alfabético en terminología castellana pensando en dar las máximas facilidades posibles al usuario. Por esta razón, se ha incluido también un apéndice de traducción inglés-castellano para los menos avezados.

Un detalle curioso es que, en previsión de los continuos cambios que experimenta el mundo de la informática, al final de cada capítulo hay una página en blanco para añadir lo que sea menester.

En resumen, un libro que no debe faltar en el bolsillo de cualquier principiante o entusiasta de los temas informáticos.

Microfile

José Manuel LAZO

BASE DE DATOS PARA EL SISTEMA MDS DE PIN SOFT

Hace unas semanas comentamos en la sección Expansión el sistema de manejo de ficheros diseñado por PIN SOFT para el microdrive de Sinclair. Como aplicación del mismo hemos creado una base de datos capaz de manejar un total de 85 K's de datos a la vez.

El programa en cuestión usa ese sistema para manejar los registros del archivo por lo que es necesario tener cargado el MDS en la memoria del ordenador previamente a la ejecución de la base de datos.

Una vez hayamos cargado también la base de datos en nuestro ordenador el programa espera a que introduzcamos el cartucho de datos, y, una vez hecho pulsaremos cualquier tecla con lo que accederemos al menú principal.

A partir de este momento usaremos el microdrive para almacenar los datos que tecleemos, teniendo esto la ventaja de que si por un casual se produce un corte de luz, no perderemos todo el trabajo que hayamos hecho, ya que en la memoria RAM de nuestro ordenador no existirá prácticamente ningún dato del fichero.

Si vamos a usar el programa por primera vez es necesario que efectuemos unas operaciones previas sobre el cartucho que vayamos a utilizar como almacenamiento de los datos. Primero, es imprescindible un formateado del mismo con la orden FORMAT «m»;1; «Fichero» operación ésta que tarda, como ya habréis tenido oportunidad de comprobar, una media de 1 minuto.

A continuación, con el MDS ya cargado en la memoria, es preciso reservar el espacio suficiente dentro del cartucho para el archivo. Esto lo hacemos con la orden RND FORMAT «Fichero»;1;85, que tarda ya un tiempo considerable, 5 minutos, dado que hay que reservar 85 K's de datos en el cartucho.

Esto sólo hay que hacerlo la primera vez para disponer del microdrive limpio de datos para empezar a usar la base de datos.

EL MENU

Se nos presenta un menú con 7 opciones, vamos a verlas una a una:

1. Introducir fichas. Es fundamental en todo fichero el poder introducir información dentro de él, en éste esta operación la realizamos con esta opción del menú. Cuando hayamos pulsado la opción se nos preguntará a partir de qué ficha deseamos empezar a introducir fichas pudiendo responder con un número dentro del rango 0 a 3125.

A partir de este momento empezaremos a entrar los textos o números correspondientes a cada ficha teniendo en cuenta que si la ficha ya estuviera escrita, la base de datos nos lo comunica ofreciéndonos dos opciones: pasar a la siguiente ficha o sobreescribir la misma.

El texto de cada registro ha de ser como máximo de 27 caracteres, aunque si ponemos más lo único que ocurre es que sólo se van a utilizar los 27 primeros. Esto viene indicado por una raya vertical que nos marca el último carácter que se puede introducir.

Entre ficha y ficha se nos ofrece la opción de vuelta al menú o introducir la siguiente ficha.

2. Ver fichas. Esta opción se encarga de presen-

```
  10 BORDER 0: PAPER 0: INK 4: POKE 23624,4: CLS
  15 PRINT "Introduzca cartucho fichero""Y pulse una tecla": PAUSE 0
  20 RND OPEN #"Fichero";1;20
  30 REM Menu
  40 LET linea=5: CLS : PRINT AT linea,5;"1. Introducir fichas"
  50 PRINT AT linea+2,5;"2. Ver fichas"
  60 PRINT AT linea+4,5;"3. Salir del programa"
  70 PRINT AT linea+6,5;"4. Back-up"
  80 PRINT AT linea+8,5;"5. Borrar fichas"
  90 PRINT AT linea+10,5;"6. Reajustar fichero"
 100 PRINT AT linea+12,5;"7. Trasferir fichas"
 110 LET b$=INKEY$: IF b$="" THEN GO TO 110
 120 IF b$="3" THEN GO TO 870
 130 IF b$="4" THEN GO TO 840
 140 IF b$="2" THEN GO TO 680
 150 IF b$="1" THEN GO TO 200
 160 IF b$="5" THEN GO TO 330
 170 IF b$="6" THEN GO TO 430
 180 IF b$="7" THEN GO TO 920
 190 GO TO 110
 200 REM Entrar fichas
 210 CLS
 220 INPUT "Primera ficha? ";primera
 230 DIM a$(27)
 240 POINT AT primera
 250 RND INPUT a$: IF CODE a$(1)<>13 THEN GO TO 630
 260 DIM a$(27): POINT AT primera: INPUT "Texto de la ficha      |"; LINE a$
 270  RND PRINT a$
 280 PRINT #0;"Otra ficha  Menu"
 290 LET b$=INKEY$: IF b$="" THEN GO TO 290
 300 IF b$="O" OR b$="o" THEN LET primera=primera+1: GO TO 230
 310 IF b$="M" OR b$="m" THEN GO TO 30
 320 GO TO 290
 330 REM Borrar fichas
 340 CLS : INPUT "Cual es la primera ficha? ";primera
 350 INPUT "Cual es la ultima? ";ultima
 360 FOR a=primera TO ultima
 370 DIM a$(27)
 380 LET a$(1)=CHR$ 13
 390 POINT AT a
 400 RND PRINT a$
 410 NEXT a
 420 GO TO 30
 430 REM Reajustar
 440 CLS : PRINT "Esta opcion tarda bastante        Ten paciencia, hay 3125 fichas""'"Pulsa 'SPACE' para dejar el       reajuste"
 450 LET ajuste=0
 460 FOR a=0 TO 3124
 470 IF INKEY$=" " THEN GO TO 610
 480 PRINT AT 5,0;"Estoy ajustando la ficha ";a
 490 DIM a$(27)
 500 POINT AT a
 510 RND INPUT a$
 520 IF CODE a$(1)=13 THEN NEXT a
 530 IF a=ajuste THEN LET ajuste=ajuste+1: NEXT a
 540 POINT AT ajuste: LET ajuste=ajuste+1
 550 RND PRINT a$
 560 POINT AT a
 570 DIM a$(27)
 580 LET a$(1)=CHR$ 13
 590 RND PRINT a$
 600 NEXT a
 610 IF INKEY$=" " THEN GO TO 610
 620 PRINT : PRINT : PRINT "Fichero ajustado"'"Tiene ";ajuste;" fichas.": PAUSE 0: PAUSE 0: GO TO 30
 630 REM Sobreescritura
 640 INPUT "Ficha ya escrita. Quieres sobre-escribirla (s/n)"; LINE b$
 650 IF b$="s" OR b$="S" THEN GO TO 260
 660 IF b$<>"n" AND b$<>"N" THEN GO TO 640
 670 GO TO 280
 680 REM Ver fichas
 690 INPUT "Cual es la primera? ";primera
 700 CLS : FOR a=0 TO 20
 710 IF primera+a>3125 THEN GO TO 780
 720 POINT AT primera+a
 730 DIM a$(27)
 740 RND INPUT a$
 750 LET c$=STR$ (primera+a): LET c$="000"( TO 4-LEN c$)+c$
 760 PRINT AT a,0;c$;" "; INVERSE 1;a$ AND CODE a$(1)<>13
 770 NEXT a
 780 PRINT #0;"Otra pagina  Menu  Impresora"
 790 LET b$=INKEY$: IF b$="" THEN GO TO 790
 800 IF b$="o" OR b$="O" THEN LET primera=primera+21: CLS : GO TO 700
 810 IF b$="m" OR a$="M" THEN GO TO 30
 820 IF b$="i" OR b$="I" THEN COPY
 830 GO TO 790
 840 REM Back-up
 850 CLEAR : INPUT "Version? "; LINE a$: SAVE *"m";1;a$ LINE 0
 860 GO TO 30
 870 REM Salida
 880 CLS : PRINT "Pulsa 's' para salir"
 890 PAUSE 0: IF INKEY$<>"s" AND INKEY$<>"S" THEN GO TO 30
 900 RND CLOSE #
 910 RANDOMIZE USR 0
 920 REM Trasferir fichas
 930 CLS : INPUT "Ficha origen? ";origen
 940 INPUT "Ficha destino? ";destino
 950 INPUT "Numero de fichas? ";numero
 960 LET desplazamiento=destino-origen
 970 FOR a=origen TO origen+numero-1
 980 POINT AT a
 990 DIM a$(27)
1000 RND INPUT a$
1010 POINT AT a+desplazamiento
1020 RND PRINT a$
1030 PRINT AT 0,0;"Transfiero ficha numero ";a
1040 NEXT a
1050 GO TO 30
```

tarnos en pantalla el contenido de las fichas que le marquemos. A la izquierda se indicará el número de ficha, y a la derecha, el texto de la misma. Cuando la pantalla estuviera llena de fichas podemos pasar a la siguiente página, sacar una copia por impresora o volver al menú.

De igual manera que con la opción anterior hay que indicar la primera ficha que queremos ver entre la 0 y la 3.125.

3. Salir del programa. La filosofía con la que trabaja esta base de datos es tener una ficha en la memoria del ordenador y acceder al microdrive sólo en el caso de que esto fuera preciso. Las últimas fichas que hayamos introducido se han de grabar en el caso de que deseáramos terminar nuestra labor antes de desenchufar el ordenador, para ello usaremos esta opción.

4. Back-up.. De todos es sabido que es más que interesante tener back-up's de seguridad de nuestros programas, pues bien, esta opción nos sirve para volcar la base de datos en un cartucho virgen de forma que luego la podamos volver a cargar con el uso de RUN + ENTER.

5. Borrar fichas. Hay veces que nos interesa el borrar un bloque de información del fichero, esta opción nos lo pone muy fácil debiendo darle el dato de la primera ficha que deberemos borrar y el de la última sobre la que se va a ejercer la función de borrado.

6. Reajustar fichero. El fichero tiene una capacidad, como ya habréis podido daros cuenta, de 3.126 fichas de 27 caracteres cada una. Es obvio que podemos poner información en cualquier parte del fichero, pero es muy probable que en un momento dado os interese disponer de toda aglutinada en las fichas inferiores, pues bien, esta opción del menú ofrece esta cualidad.

Aunque, todo hay que decirlo, esta operación es lenta, piénsese que es mucha información para nuestro Spectrum la que tiene que manejar y deberemos de hacerla sólo en el momento en que podamos dejar al ordenador trabajar solo unos minutos.

De todas formas, si deseáramos dejar el reajuste de fichas en cualquier momento bastará pulsar la tecla «Space». En la pantalla se nos informará en todo momento de la ficha que es objeto de ajuste en ese instante.

7. Transferir fichas. Esta última opción nos ofrece un interesante comando que sirve para copiar bloques de fichas en otra posición del fichero. Habrá que responder a la base de datos con las preguntas que nos hará sobre cuál es la primera ficha que hay que copiar, dónde vamos a situar el bloque y cuántas fichas tiene como longitud el bloque copiado.

De igual manera que con la opción 6 del menú, se nos tendrá informados en todo momento de qué ficha es la que es objeto de desplazamiento.

PARA FINALIZAR

Trabajar con esta base de datos es muy cómodo ya que esto es uno de los principales objetivos que se han perseguido en su confección y salvo las opciones 6 y 7 del menú, que se toman su tiempo, se observará una gran rapidez de proceso si se tiene en cuenta que se trabaja con microdrive.

No se ha incorporado ningún tipo de algoritmo de ordenación del fichero debido a que las pruebas que se realizaron en este sentido resultaron totalmente inviables por causa del excesivo tiempo que consumían.

El programa está enteramente escrito en Basic debido a que no ha sido precisado la utilización de ninguna rutina en CM, salvo por supuesto, el sistema MDS, pero de esto no deberemos de preocuparnos dado que sólo hay que cargarlo y ya olvidarnos de él, la base de datos se ecargará de su gestión.

LE CONCEDEMOS TRES DESEOS



CONSULTORIO

«RANDOMIZE USR...»

Quiero intentar poner vidas infinitas al «Camelot Warriors» con un «POKE» especial de MICROMANIA, concretamente éste:

POKE 50782,255

El problema que tengo es que, en este apartado de vuestra revista, dice que hay que hacer MERGE "" ENTER, después del O.K. POKE 50782,255 antes del RANDOMIZE USR y luego ENTER. Pero yo sé que ese USR necesita otro número detrás y no sé cuál es, porque no tengo los suficientes conocimientos de Basic y C/M. ¿Me podrían decir cómo se averigua ese número?

Victor SANCHEZ · Badajoz

☐ La mayoría de los programas comerciales, llevan un cargador en Basic que incluye, al menos, las siguientes líneas:

10 LOAD "" CODE
20 RANDOMIZE USR ...

Los números de línea pueden variar o pueden aparecer más líneas, pero básicamente, estas dos deben estar. En la línea 20 y después del «USR» está la dirección de arranque del programa, el «POKE» hay que meterlo después de la instrucción «LOAD» y antes de la «RANDOMIZE USR ...». En este caso, sería:

15 POKE 50782,255

Para arrancar el programa, deberá hacer «GO TO 1» y poner en marcha, de nuevo el cassette.

Cuestión de bytes

¿Con qué otros ordenadores, además del Spectrum y del ZX-81, es compatible la impresora GP-50,S?

¿Cuál de las siguientes sentencias ocupa menos memoria?

1. LET A = VAL «0»: LET B = VAL «2»
2. LET A = LEN "": LET B = LEN «AY»
*3. LET A = NOT PI: LET B = NOT A*2*

Carlos E. CARLEOS · Asturias

☐ La impresora GP-50,S no es compatible con ningún ordenador más, ni siquiera, con el «QL».

Le vamos a decir la longitud de cada sentencia y usted decide cuál usar:

1.	LET A = VAL «0»	7 bytes
	LET B = VAL «2»	7 bytes
2.	LET A = LEN ""	6 bytes
	LET B = LEN «AY»	8 bytes
3.	LET A = NOT PI	5 bytes
	LET B = NOT A*2	13 bytes

Nosotros le aconsejamos la siguiente:

LET A = NOT PI: LET B = VAL «2»

Que ocupa 13 bytes, incluidos los dos puntos.

«DUMP»

En el número 71 de su revista hay un programa llamado «MARTE» y dice que hay que realizar un «DUMP» en la dirección 47300.

Me gustaría saber cómo obtener el «DUMP» y qué hay que hacer para realizarlo en determinada dirección.

Carlos E. DE ANDRES · Valencia

☐ El programa «MARTE» consta de dos listados; el primero está en Basic y tiene tres líneas (por cierto que hay un pequeño error, la primera línea debe ser la número 0, no la número 1) el segundo listado está en el formato del CARGADOR UNIVERSAL DE CODIGO MAQUINA, por tanto, deberá teclearlo utilizando este programa. Una vez introducidas todas las líneas, pulse «ENTER» y le saldrá un pequeño menú en la parte inferior de la pantalla. Ahora pulse «D» y, cuando le pregunte en qué dirección, responda 47300. Cuando termine el «DUMP»,utilizando la opción SAVE (OBJETO) grábalo a continuación del Básic, indicando 47300 como dirección de inicio y 4739 como número de bytes.

Las Rutinas de la ROM

Me ha parecido una buena idea el publicar, en forma de fichas, las rutinas de la ROM. Sin embargo, creo que os habéis excedido en lo esquemático de las explicaciones.

En la rutina «FP-CALC» (28h), ¿cómo se introducen los números?, ¿cómo se sacan los resultados del stack del calculador?

En la rutina «BEEPER» (03B5h) no entiendo cómo se componen los registros «DE» y «HL» para darlos como datos de entrada. Podríais dar como ejemplo la forma de hacerlo para la nota «DO» de forma que dure un segundo.

Por último, quiero agradeceros el truco de colocar confetis en el teclado del Spectrum Plus, ya que a mí me funciona perfectamente.

Manuel CAÑETE · Barcelona

☐ Tanto respecto a las rutinas de la ROM, como a los restantes temas, las fichas tienen la utilidad de servir como medio de consulta rápida para quien ya domine el código máquina. Todos estos temas, se tratan con más profundidad dentro del curso de código máquina que se publica simultáneamente, de forma que, el curso y las fichas, constituyan herramientas complementarias para el programador.

Para utilizar las rutinas de la ROM es necesario saber bastante de código máquina, por ello, este tema se tratará en el curso cuando se hayan terminado de ver todas las instrucciones.

No obstante, y para no hacerle esperar, le anticiparemos algo. La rutina del calculador (FP-CALC) trabaja de forma similar a una calculadora de altas prestaciones que utilice la «notación polaca inversa»; esta notación es la utilizada por las calculadoras «Hewlet Packcard» y exije que los dos datos a operar se hallen en el «stack» antes de realizar la operación . Quienes están acostumbrados a manejar estas calculadoras, aseguran que la notación polaca inversa es la más cómoda de usar.

Para introducir y sacar datos del stack, tenemos otra serie de rutinas en la ROM y para indicar las operaciones a realizar, utilizaremos literales colocados a continuación de la instrucción «RST * 28». De todas formas, el manejo del calculador es bastante complicado para explicarlo aquí. Pero prometemos dedicarle un capítulo entero en el curso de código máquina para aprender a sacarle el máximo partido.

Respecto a la rutina «BEEPER», ésta es más fácil de utilizar. Llamemos «f» a la frecuencia de la nota expresada en ciclos por segundo y «t» a la duración en segundos. Las fórmulas para obtener los contenidos de «DE» y «HL» son las siguientes:

«DE» = INT (f*t)
«HL» = INT ((t*6689/4)—30.125)

La tabla de frecuencias para todas las notas, puede encontrarla en el rever-

so de la micro-ficha M-8. Recuerde que para subir una octava, deberá multiplicar la frecuencia por dos, y dividirla por dos para bajar una octava.

En el caso concreto que nos plantea, «f» vale 261.63 y «t» vale 1, por tanto, «DE» contendrá INT (261.63 • 1) = = 261d (Ø1Ø5h) y «HL» contendrá INT ((1 • 6689/4)—3Ø.125) = 1642d (Ø66Ah).

«Sir Fred»

¿Podrían decirme cómo se pasa la cuarta pantalla del «Sir Fred»? Cuando llego aquí me tiro al agua y, al intentar subir donde está ese señor de nariz gorda, el muñeco no sube.

Felipe RODRIGUEZ - Madrid

☐ El señor de la nariz gorda es un borracho, por tanto, lo que quiere es la botella. Como es un tipo muy amable, le permitirá usar la barca a cambio.

El Cargador Universal de código máquina

En el cargador universal de código máquina en la zona de LOAD, hay una instrucción que no sé qué función tiene. La instrucción es: RANDOMIZE USR 23296.

Enrique GUZMAN - Madrid

☐ El cargador de código máquina guarda el código fuente en una variable de cadena. Cuando se quiere salvar el código fuente en cassette, se hace como si se tratase de una matriz, es decir, con «SAVE...DATA» pero al intentar cargarlo de nuevo, nos encontramos con el problema de que se carga como una matriz y no podemos acceder a ella. Para revolver esto, se utiliza una pequeña rutina en código máquina que está situada en el buffer de impresora, es decir, a partir de 23296. Esta rutina cambia el identificador de la variable para que vuelva a ser accesible.

¡NOS CAMBIAMOS!

Queremos comunicar a nuestros «seguidores» incondicionales que nos cambiamos de dirección que será, a partir de ahora, ésta que os ofrecemos:
Carretera Madrid-Irún kilómetro 12,400.
28049 Madrid.
Teléfonos: 734 70 12 (redacción)
734 65 00 (suscripciones)

DE OCASION

● VENDO copiador TC-7 por 600 ptas. Antonio Márquez c/ San Pedro, 73. Puertollano (Ciudad Real).

● NECESITO las instrucciones completas y claras de Omnicalc 2, Gens 3M, Mons 3M, VU-3D, Deux ex Machina, Melbourne Draw, 3D Starstrike, Fighter Pilot y Cobal. A cambio regalo libros de C.J. Cela, Alberti, Henry Miller, etc. o bien pago gastos de envío. José Manuel Vizaino c/ Emilia Pardo Bazán, 10 4.°. Lugo. Tel (982) 22 97 09.

● VENDO olivetti M-24 de 128 K con dos drives para diskettes de 360 K cada uno, una impresora Olivetti PR-1450 compatible, así como diskettes de aplicación, textos y cálculos. Todo nuevo y en perfecto estado por sólo 34.000 ptas. Interesados llamar al Tel. (983) 35 09 73. Valladolid. Preguntar por Rafael Díez.

● CAMBIO bicicleta Phanter de Cross con marchas, amortiguación en buen estado, por un ordenador Spectrum Plus o 48 K. Interesados llamar al Tel. (91) 206 32 94. Preguntar por Gelo.

● VENDO Spectrum 48 K con todos los accesorios más interface programable para joystick Quick Shot III, con teclado incorporado. Precio: 39.000 ptas. Interesados llamar al Tel. (91) 729 31 99.

● VENDO Zx Spectrum 48 K, con adaptador, cables y manual de instrucciones, en perfecto estado de funcionamiento. Todo por sólo 28.500 ptas. Interesados contactar con Carlos Rodríguez. Avda. Almería, BL-7, 3.° 1.ª. Badalona (Barcelona). Tel. (93) 398 54 82.

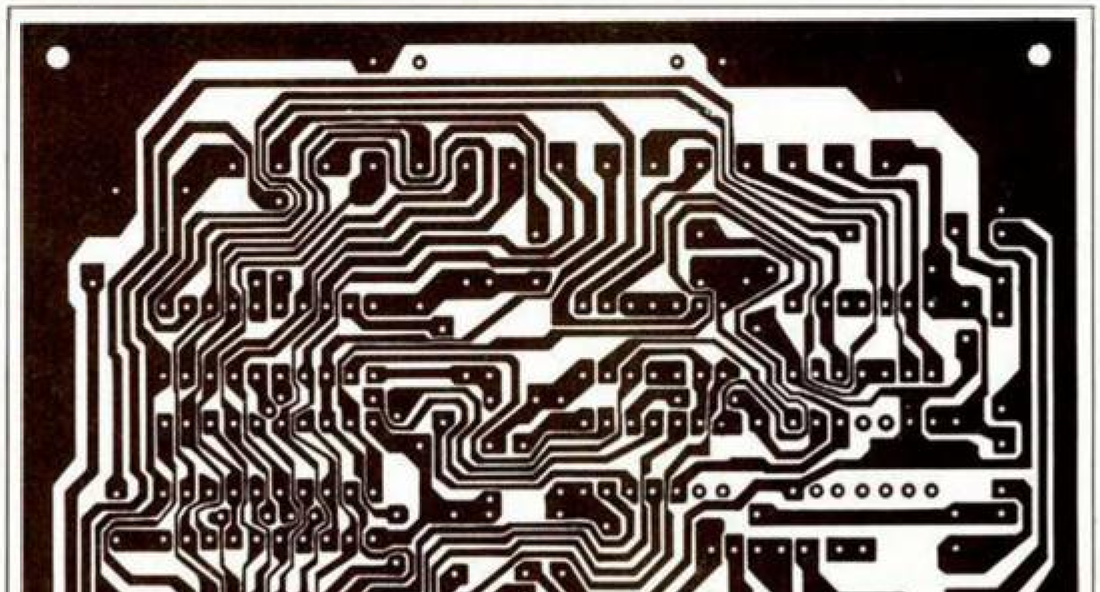
● VENDO Spectrum 48 K, con teclado Indescomp, amplificador de sonido, interrruptor on/off y joystick (cursores incorporados), en garantía. Precio a convenir. Preguntar por Yon a partir de las 6 tarde. Tel. (943) 39 63 36. San Sebastián.

● VENDO Spectrum 48 K, fuente de alimentación, manual, libro de código máquina todo por 30.000 ptas. Vendo interface 1, microdrive, por 23.000 ptas. También consola de videojuegos Atari, por 25.000 ptas. Interesados en la oferta dirigirse al Tel. 319 35 41 de Barcelona. Preguntar por Hugo (noches).

● VENDO teclado original del Spectrum 48 K por 10.000 ptas. Interesados llamar al Tel. 742 81 73 de Madrid (7 a 10 noche). Preguntar por Alfredo.

● VENDO Spectrum 48 K en perfecto estado, comprado en enero-85, intrucciones en castellano, regalo la cinta Horizontes también en castellano, sólo por 19.500 ptas. Llamar al Tel. (93) 235 62 38. Preguntar por Manuel.

Ante las innumerables cartas recibidas solicitándonos nuevamente la publicación de la placa del Grabador de Eprom, no hemos tenido más remedio que acceder a ello ya que, por defectos en la impresión de la otra (MICROHOBBY n.° 37) los detalles de construcción de la misma no quedaban suficientemente claros.



Dibujo de la cara de pistas del grabador de EPROM, tamaño real.


SPECTRUM
COMMODORE
AMSTRAD
BOMB JACK
ZAFI CHIP
¡INSTRUCCIONES EN CASTELLANO!
Después del tremendo éxito del programa en Japón, Bomb Jack ha pasado a ser el programa de arcade más famoso en USA y muy pronto lo será en toda EUROPA.
¡Diviértete!
Si están agotados
habitual
ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141. 28046 Madrid.
Tel. 459 30 04. Tel. Barna. 209 33 65.
Telex: 22690 ZAFIR E
elite



SPECTRUM 128
EL SUMMUM

Spectrum, como líder, marca un nuevo hito en la historia de los ordenadores familiares.

El Spectrum 128.

Gran capacidad de memoria. Teclado y mensajes en castellano, teclado independiente para operaciones numéricas y de tratamiento de textos...

Sinclair e Investrónica han desarrollado una auténtica novedad. En ningún lugar del mundo, salvo en los Distribuidores Exclusivos de Investrónica, podrás encontrar el nuevo Spectrum 128.

Sé el primero en tener lo último.

SPECTRUM 128. NOVISSIMUS

DISTRIBUIDOR EXCLUSIVO

investronica

Tomás Bretón, 62.
Tel. (91) 467 82 10.
Telex 23399 IYCO E.
28045 Madrid

Camp, 80.
Tels. (93) 211 26 58 - 211 27 54.
08022 Barcelona